AFRL-RX-TY-TR-2011-0005

JOINT COLLABORATIVE TECHNOLOGY EXPERIMENT (JCTE) FINAL REPORT

Rodney L. Brown

Mesa Robotics
9238 Madison Boulevard
Building 2, Suite 116
Madison, AL 35758

Contract No. FA4819-09-C-0045

December 2010

DISTRIBUTION A: Approved for public release; distribution unlimited.
88ABW-2013-2501, 28 May 2013

AIR FORCE RESEARCH LABORATORY
MATERIALS AND MANUFACTURING DIRECTORATE

■ **Air Force Materiel Command** ■ **United States Air Force** ■ **Tyndall Air Force Base, FL 32403-5323**

DISCLAIMER

NOTICE AND SIGNATURE PAGE

AFRL-RX-TY-TR-2011-0005 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

//SIGNED//
WALTER M. WALTZ
Work Unit Manager

//SIGNED//
DEBRA L. RICHLIN
Acting Chief, Airbase Engineering Development Branch

//SIGNED//
ALBERT N. RHODES, PhD
Chief, Airbase Technologies Division

REPORT DOCUMENTATION PAGE			*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
31-DEC-2010	Final Technical Report	01-OCT-2008 -- 31-MAY-2010

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Joint Collaborative Technology Experiment (JCTE) Final Report	FA4819-09-C-0045
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
	0909999F
6. AUTHOR(S)	5d. PROJECT NUMBER
Brown, Rodney L.	GOVT
	5e. TASK NUMBER
	F0
	5f. WORK UNIT NUMBER
	QF503021

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Mesa Robotics 9238 Madison Blvd, Bldg 2, Suite 116 Madison, AL 35758	

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
Air Force Research Laboratory Materials and Manufacturing Directorate Airbase Technologies Division 139 Barnes Drive, Suite 2 Tyndall Air Force Base, FL 32403-5323	AFRL/RXQ
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)
	AFRL-RX-TY-TR-2011-0005

12. DISTRIBUTION/AVAILABILITY STATEMENT

Distribution Statement A: Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES

Ref Public Affairs Case # 88ABW-2013-2501, 28 May 2013. Document contains color images.

14. ABSTRACT

This report will provide historical background, summarize year one results for the JCTE project, and outline a path forward for follow-on work. JCTE is a proof of principle effort funded by the Joint Ground Robotics Enterprise (JGRE). Year one focused on integration of systems from the partner organizations, development of unmanned systems collaborative behaviors, a number of integration/test sessions, simulation of multi-vehicle collaboration, and concluded in October 2008 with a technical capabilities demonstration. Year two JCTE efforts focused on hardware and software refinements to increase reliability, robustness, and user friendliness, and additional collaborative behaviors to further enhance capabilities and reduce workload. The experiment demonstrated the possibilities of ground and air marsupial unmanned systems deployed from a UGV to conduct route clearance IED defeat operations.

15. SUBJECT TERMS

Defender, RMAX, Joint Collaborative Technology Experiment (JCTE), Joint Ground Robotics Enterprise (JGRE), robotics, UAV, UGV, integration, Joint Capability Areas (JCA), UAS, JAUS, C4ISR, HMMWV, Mongoose, Multi-robot Operator Control Unit (MOCU), AUMS, MDARS, LMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Walter M. Waltz
U	U	U	UU	107	19b. TELEPHONE NUMBER *(Include area code)* 850-283-3725

Reset

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

PREFACE

Use of unmanned systems is rapidly growing within the military and civilian sectors in a variety of roles including reconnaissance, surveillance, force protection and perimeter security. As utilization of these systems grows at an ever increasing rate, the need for unmanned systems teaming and inter-system collaboration between unmanned systems becomes apparent. Collaboration provides the means of enhancing individual system capabilities through relevant data exchanges that contribute to cooperative behaviors among systems and enables new capabilities which are not possible if the systems operate independently. A collaborative, networked approach to development holds the promise of adding mission capability, while simultaneously reducing the workload of system operators. The Joint Collaborative Technology Experiment (JCTE) joins individual collaborative technology development efforts within the Air Force, Army, and Navy to demonstrate the potential benefits of interoperable multiple system collaboration in a force protection application. JCTE participants are the Air Force Research Laboratory, Materials and Manufacturing Directorate, Airbase Technologies Division, (AFRL/RXQ), the Army Aviation and Missile Research, Development and Engineering Center, Software Engineering Directorate (AMRDEC SED), and the Space and Naval Warfare Systems Center – Pacific, Unmanned Systems Branch (SSC-Pacific). The Robotics JCTE Team at AFRL/RXQ consisted of personnel from Applied Research Associates, Inc.; Wintec Inc.; MESA Robotics; and AFRL/RXQ engineers.

1. EXECUTIVE SUMMARY

This report will provide historical background, summarize year one results for the Joint Collaborative Technology Experiment (JCTE) project, and outline a path forward for follow-on work. JCTE is a proof-of-principle effort funded by the Joint Ground Robotics Enterprise (JGRE). Year one focused on integration of systems from the partner organizations, development of unmanned systems collaborative behaviors, a number of integration/test sessions, simulation of multi-vehicle collaboration, and concluded in October 2008 with a technical capabilities demonstration. The demonstration was conducted at the Silver Flag test facility located at Tyndall Air Force Base (AFB), FL. The Silver Flag demonstration brought together two unmanned air systems (UAS), three unmanned ground vehicles (UGVs), a beyond line-of-sight (BLOS) command and control (C2) capability, and a variety of collaborative behaviors allowing a small number of system operators to detect and engage a simulated hostile threat at a remote airfield five miles from the operations center. The unmanned systems all utilized a common communications protocol to enable collaboration and a common operator interface and C2 system for maximum operator situational awareness.

Year two JCTE efforts focused on hardware and software refinements to increase reliability, robustness, and user friendliness, and additional collaborative behaviors to further enhance capabilities and reduce workload. The second-year effort culminated in a Warfighter Experiment that utilized an improvised explosive device (IED) scenario taken from requirements of Air Combat Command (ACC), the Air Force Civil Engineer Support Agency (AFCESA) and the explosive ordnance disposal (EOD) community users. The experiment took place at Test Range 3, Tyndall AFB, and demonstrated the possibilities of ground and air marsupial unmanned systems deployed from an unmanned ground vehicle to conduct route clearance improvised explosive device (IED) defeat operations.

2. INTRODUCTION

Unmanned air and ground systems utilization by military forces has grown dramatically in the past 5 years. In October of 2000, US Congress passed Public Law 106-398, the National Defense Authorization Act for FY2001, which established goals for the fielding of unmanned systems. The goals are “by 2010 one-third of the aircraft in the operational deep strike force aircraft fleet are to be unmanned,” and “by 2015 one-third of operational ground combat vehicles are to be unmanned.” At the start of Operation Iraqi Freedom (OIF) U.S. military forces had fewer than 200 operational UAS’s. As of November 2008, there were more than 6,000 UAS’s deployed in support of OIF and Operation Enduring Freedom (OEF). UAS’s flew approximately 400,000 hours in 2008 in support of these operations. As of December 2010, there are currently more than 4,000 UGSs deployed in theater.

This recent real-world experience with fielded unmanned systems has shown that these systems can provide significant value added in a wide variety of roles. New concept of operations (CONOPS) and new tactics, techniques, and procedures (TTPs) are being explored continuously. Among the lessons learned from these operations, common complaints being fed back from users to the research and development (R&D) community include a lack of inter-operability between systems and a lack of system autonomy, resulting in high operator workload.

Inter-vehicle collaboration provides the means to address these shortcomings. Collaborative behavior as applied to unmanned systems is defined as two or more unmanned systems working together to accomplish predefined mission(s) with minimal human operator intervention. It is important to differentiate between a scenario in which multiple unmanned systems utilize inter-system collaboration and other multi-vehicle scenarios. Significant characteristics of multi-vehicle collaboration include the ability for unmanned systems to work as a team, command one another, pass information directly to each other, and make changes to their missions based on that information while being monitored by a human operator. In a non-collaborative environment multiple vehicles operate independently of one another, usually require one or more operator per system, all sensor data is fed back to the operator(s), and all mission decisions are made by the operators. This non-collaborative environment imposes a high level of workload on operators and requires a tremendous amount of coordination between them to accomplish even mundane tasks. This high-workload environment may contribute to a loss of situational awareness for battle commanders.

JCTE goals are to develop, refine, integrate, and demonstrate collaborative technology enablers that address needs within multiple Joint Capability Areas (JCAs). JCTE will provide enabling technologies to directly support the following JCAs and tier 2 capability areas:

- Land operations—Joint fires, small-unit support, weaponization, navigation, cross-country mobility
- Force Protection—Counter IED, physical security, explosive ordnance disposal (EOD), counter sniper
- Special operations—Tactical mission support
- Battlespace awareness—Persistent Intelligence, Surveillance and Reconnaissance (ISR)

The mission scenario for JCTE is a remote site security application that highlights the capabilities of the component technologies the three partner organizations bring to the project. The JCTE scenario will demonstrate BLOS C2 of multiple heterogeneous unmanned systems, collaborative roving perimeter patrols by multiple unmanned systems, persistent close-in aerial surveillance by a vertical takeoff and landing (VTOL) UAS supported by in-the-field autonomous launch, recovery and refueling by a UGV, collaborative target ID and lethal engagement, and post engagement analysis. The scenario requires a high level of interoperability between multiple heterogeneous unmanned systems, enhanced unmanned systems capabilities through the application of collaboration, and a relatively low operator workload given the number of systems employed.

A basic enabling requirement for inter-system collaboration is the use of a standardized communications protocol for C2 of unmanned systems. To date the unmanned systems industry and the R&D community have not arrived at a consensus for standardization of communications. As a result most unmanned systems utilize proprietary C2 schemes. Efforts toward standardization have resulted in the development of the two best-known protocols, Standardization Agreement (STANAG) 4586, and the Joint Architecture for Unmanned Systems (JAUS). STANAG 4586 was developed to support UAS operations and has gained some acceptance with UAS developers and the UAS user community. However, STANAG 4586 was not intended to support unmanned systems operating in other domains (e.g., UGVs). Use of a UAS-specific communications protocol inhibits the ability of a UAS to collaborate across domains with a UGV or unmanned surface vehicle (USV). This is problematic in a scenario utilizing tactical assets that could benefit from collaboration, for example a force protection application utilizing a small UAS such as a Raven and a Mobile Detection Assessment and Response System (MDARS) UGV. JAUS was developed for use with unmanned ground systems, but from the start was intended to support operations in all domains (land, air, surface, and sub-surface). At this point in time the JAUS protocol provides the best means of multi-vehicle inter-operability across multiple domains so it was chosen as the protocol for all unmanned systems and operator interfaces for JCTE. JAUS has been adopted by the Society of Automotive Engineers (SAE) under Aerospace Standard -4 (SAE AS-4).

3. BACKGROUND

The three JCTE partner organizations all have a long history in research and development of unmanned systems. All have an extensive background with JAUS through utilization in a variety of on-going projects and participation in SAE AS-4 working group meetings. AMRDEC SED is a leader in the areas of remote engagement and automated lethality and in the use of simulation in virtual unmanned systems development, and is the initial developer of the JAUS protocol. The Robotics JCTE Research Team of the AFRL/RXQ conducts advanced research and development of intelligent unmanned systems. AFRL's primary research areas include advanced robotics technologies development with focus on intelligent systems, EOD, automated range clearance, first responders, aircraft and airbase operations support systems, and force protection. SSC-Pacific provides network-integrated robotic solutions for command, control, communications, computers, intelligence, surveillance, and reconnaissance (C4ISR) applications supporting UGVs, UASs, USVs, unattended ground sensors (UGS) and unattended weapons systems. SSC-Pacific conducts R&D in unmanned systems supporting a wide variety of applications including EOD/IED detect and defeat, and force protection.

The JCTE project began with individual developmental efforts at the three partner organizations in the early 2000s. These efforts had a number of common threads—all were Joint Robotics Program (JRP, the predecessor to JGRE) funded, all incorporated some level of multi-vehicle collaboration and all utilized the JAUS protocol for C2. SSC-Pacific was developing capability to launch, recover, refuel and transport small VTOL UASs on a UGV. AFRL was developing an airborne communications link to extend the operational range of UGVs beyond line of sight. AMRDEC was developing JAUS messages specifically to support collaborative operations for multi-vehicle teaming, and capabilities for multi-vehicle collaboration to conduct lethal fires.

In 2005 the three labs merged these independent projects into a joint project. The JRP funded this 18-month effort, the Collaborative Engagement Experiment (CEE) in fiscal years 2005 and 2006. The goals for CEE were to demonstrate the value of collaborative behaviors in accomplishing a complex mission, and to develop a joint framework for future collaborative efforts to avoid independent Army, Air Force, and Navy solutions. CEE was to culminate in a technology demonstration in 2006 with multiple air and ground vehicles from all three organizations collaborating to conduct a lethal engagement and post engagement battle damage assessment.

Initial CEE efforts focused on maturation of the individual component technologies that the three organizations were bringing into the experiment. These capabilities, touched on in the previous paragraph, will be fully described in the next section.

CEE worked with the Soldier Battle Lab at Ft. Benning, GA, to conduct a task analysis that identified scenarios in which collaborative engagement technologies could positively impact currently defined missions. CEE researched on-going collaborative efforts Department of Defense (DoD) wide and within the academic community, meeting with users and developers to discuss collaboration scenarios, collaboration benefits, applications, and impediments to development. The task analysis and research helped to define meaningful and achievable collaboration goals for the CEE project that fit within schedule and budget constraints, and to lay out a path forward for future work.

As CEE efforts progressed it became apparent that the maturation level of the component technologies would not be sufficient to conduct the planned demonstration. CEE funding levels were insufficient to support acceleration of development efforts. The CEE partners were unable to secure additional funding in mid fiscal year 2006 to continue development efforts and conduct a modified version of the planned CEE demonstration. At the conclusion of CEE the partner organizations agreed to pursue independent funding to mature their individual component technologies, to continue to communicate and collaborate with one another, and to seek additional funding at a later date to conduct collaborative experimentation and demonstrations.

The CEE project successfully accomplished the following:

- Established a framework under which three services coordinated unmanned systems development efforts to demonstrate joint multi-vehicle collaboration in a real-world scenario.
- Researched ongoing collaborative efforts.
- Conducted user/developer meetings to establish project collaboration goals and a path forward for future work.
- Conducted a task analysis to validate value added of unmanned systems collaboration in a number of common mission scenarios.
- Expanded inter-operability of systems employed by all three labs through use of JAUS.
- Increased component technology maturity levels for all three services.

The JCTE project follows CEE after a one-year hiatus to mature component technologies. JCTE goals are an extension of the goals established for CEE—to demonstrate the value added in collaboration between multiple unmanned systems in conducting a complex mission in a real-world scenario.

4. JCTE COMPONENT TECHNOLOGIES

4.1. Multi-Robot Operator Control Unit (SSC-Pacific)

As mentioned in the Introduction, many unmanned systems today utilize C2 systems that are proprietary. This inhibits system inter-operability and potential inter-system collaboration and is also an impediment to third-party development of system upgrades, enhancements, and new capabilities. Unmanned systems researchers at SSC-Pacific experienced similar obstacles in working with a variety of ground and surface vehicles, and developed a Multi-Robot Operator Control Unit (MOCU) as a solution. The MOCU was designed for the simultaneous control of multiple, heterogeneous, unmanned systems. The MOCU operates with unmanned systems across all domains, and is not tied to a specific protocol. To date the MOCU has been used with fixed wing and VTOL UASs, various UGVs, and several different USVs.

The MOCU employs a modular, scalable, highly flexible architecture. The MOCU modularity allows control and status monitoring of multiple vehicles utilizing differing communications protocols, mapping requirements, and video codecs. It also allows for easy expansion by third parties developing new protocol modules. The MOCU achieves its modularity through the use of a fixed core module and supporting modules that provide specific functionality. Changes in a system configuration such as the addition or subtraction of unmanned assets will require the addition or subtraction of supporting modules. The core module manages data flow between modules and overall operation. The MOCU scalability allows great flexibility in hardware configuration, picking and choosing only the hardware required and appropriate for a given application. An application employing a single man-portable UGV could utilize a relatively simple hardware configuration such as a laptop with a simple joystick attached; whereas a complex multi-vehicle installation may require multiple networked computers, multiple monitors, and multiple input devices of various sorts.

The MOCU supports both control and status monitoring functions for unmanned systems. Status for all vehicles connected to the MOCU can be monitored simultaneously, but control can be exercised over only one vehicle at a time. Vehicles displayed in monitor mode appear on a geo-referenced map (Figure 1) and basic status information is displayed for each along with the option to display video from each. For a vehicle in control mode the MOCU has complete control over all vehicle and payload functions, the vehicle status is amplified, and the user interface is configured for that particular vehicle. User interfaces in the MOCU include both input devices such as joysticks, and a graphical user interface (GUI). Each is configured for each vehicle via an extensive markup language (XML) configuration file. User interfaces for different vehicles can vary dramatically and use of configuration files to manage these interfaces makes changes to system configuration relatively simple.

For these reasons the MOCU was selected as the ideal operator interface to support the JCTE. The MOCU version employed for the October demonstration was Multi-Robot Operator Control Unit 2 (MOCU2), which at the time was the most recent version, utilized for both mine warfare (MIW) and antisubmarine warfare (ASW), Littoral Combat Ship (LCS) USV projects in-house at Space and Naval Warfare Command (SPAWAR).

For the JCTE, MOCU2 was configured to have a unique GUI for each type of unmanned system employed with the GUI tailored to suit their distinctive functionalities. MOCU2 communicated with the unmanned systems via the JAUS protocol. JAUS version RA3.2 was used with the OCU and Payload Committee (OPC) 2.75 messages for dynamic discovery. For functionalities that are not yet supported in the standard version RA3.2, user-defined JAUS messages were implemented to fully support each unmanned system's capabilities. For the JCTE, MOCU2 additional functionalities were implemented to enable it to fully control the unmanned systems exercised in the demonstration. These new additions involved new implementations on three levels: MOCU2's core control routines, the GUI, and user-defined JAUS messages. For AFRL's Defender UGV, MOCU2 was updated with new functionalities for weapon pan-and-tilt and fire control, including to display weapon targets and to select target for automatic aiming. For SPAWAR's Mongoose UAS, MOCU2 extended the standard JAUS Global Waypoint Driver protocol to handle special VTOL tasks. For SPAWAR's Autonomous UAS Mission System (AUMS), MOCU2 added custom commands to center and capture the landed UAS, to inject a refuel pod, to defuel, to refuel, and to release the UAS for takeoff.

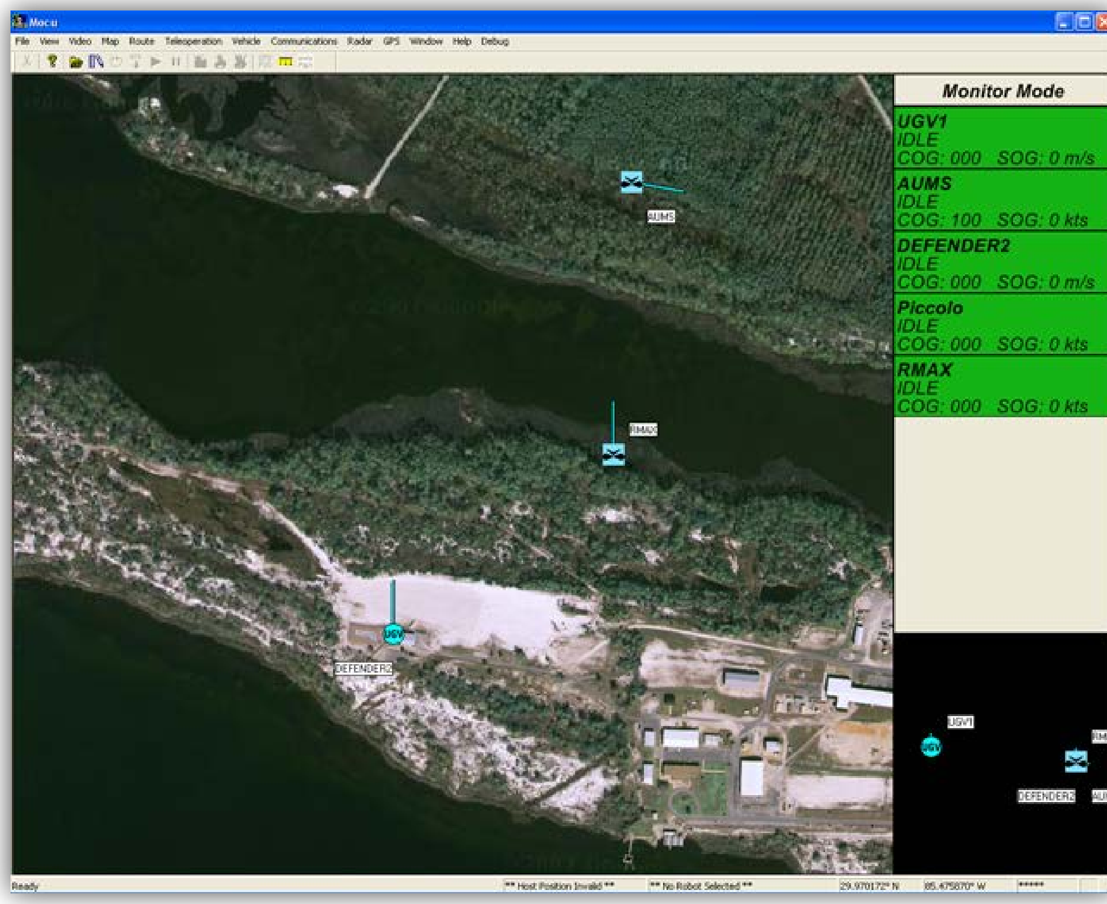


Figure 1. MOCU Screenshot in Monitor Mode with Two UGVs, Two UASs, and AUMS Connected

4.2. Autonomous UAS Mission System (AUMS) (SSC-Pacific)

4.2.1. AUMS Background

VTOL UAS can provide significant advantages over fixed wing UAS in many tactical applications. The VTOL UAS is capable of being launched and recovered in a confined or obstructed area. With appropriate sensors it can operate in a cluttered environment, can approach quite close to a target of interest, and can either hover and stare or perch and stare at a target. Primary disadvantages of the VTOL UAS are limited flight endurance, limited range, limited payload capacity, and mechanical complexity.

A significant potential advantage with a VTOL UAS is that it can land and be relaunched without the need for a human in the loop, providing a means of complete autonomous operation. This is not the case for the typical tactical fixed wing UAS. Fixed wing vehicles are typically recovered in a fashion that will not support relaunch without an operator handling the aircraft. For example, there are a number of UASs that are net recovered and catapult launched. The operator must physically remove the UAS from the landing net, service it, and then stage it in a catapult to relaunch it for another mission. This is a process that would be difficult to automate.

AUMS is a modular, vehicle borne system to autonomously launch, recover and refuel a VTOL UAS. It was designed to leverage the ability of a VTOL UAS to autonomously launch and recover and to offset the endurance, range and payload limitations via refueling in the field. AUMS can be used as a standalone system, or mounted on a ground or surface vehicle, manned or unmanned, to autonomously support one or more VTOL UASs. The AUMS mounted on a UGV provides the capability to transport a small UAS into a hazardous area and perform persistent aerial operations without endangering personnel. In a fixed installation, the AUMS provides on-demand persistent aerial operations at remote sites without the need to have personnel present at the site. The autonomous nature of the system minimizes close proximity exposure of operator personnel to perceived dangers both from the UAS and within the operational environment.

The AUMS development began in 2002 as a parallel effort for the Defense Advanced Research Projects Agency's (DARPA) Micro Air Vehicle (MAV) and Organic Air Vehicle (OAV) development efforts. MAV was intended to be small enough to be man-portable while the OAV-class vehicle was intended from the start to be transported by a ground vehicle. The AUMS was originally designed to support transport, launch, refueling, and relaunch of the OAV with sufficient flexibility and modularity to work with other sorts of VTOL UAS with minimal modification.

Initial AUMS development efforts utilized the Allied Aerospace (and, later, AAI Corporation) iStar 29i UAS and the MDARS UGV as representative examples of appropriate target platforms. A proof-of-concept demonstration was performed in 2002 with the iStar flown by a manual pilot-in-the-loop off a crude launch platform mounted on MDARS. AUMS development with the iStar proceeded for several years, but development issues with the iStar and lack of availability of an affordable ducted-fan alternative to the iStar were a significant impediment.

SSC built and tested a number of platforms for the iStar/MDARS configuration concluding with a vented platform 48 in diameter. The iStar employs a 29-in diameter ring as a landing gear base.

AUMS utilized six actuators to grasp this ring and center the UAS on the platform for capture and refueling. The first successful autonomous launches were performed with this platform and the iStar in 2005. The iStar was capable of autonomous landing on the ground but never demonstrated adequate precision landing performance to attempt an autonomous landing on the platform.

Beginning in 2005, SSC began using small helicopter UASs as surrogates for the iStar. Small helicopter UASs based on hobby-class radio control (R/C) platforms are ideal VTOL test and development platforms. Low acquisition and maintenance cost, easy integration of hardware and sensors, well understood flight dynamics, and good performance with a relatively wide performance envelope are just some of the advantages. Lessons learned in working with these helicopters should translate to other types of VTOL platforms including Lift-Augmented Ducted Fan (LADF) designs such as MAV.

4.2.2. AUMS Technical Description

AUMS is composed of five major subsystems: the launch-and-recovery platform, the refueling system, the electronics module, the air vehicle, and command and control. Design goals for the system were:

- Utilize JAUS and MOCU on AUMS, the UAS, and the UGV to support automation of the launch, recovery and refueling processes, and maximize collaboration among the three to minimize operator workload.
- Maximize landing platform size without impacting host vehicle footprint—i.e., platform should not exceed the host vehicle length or width.
- Provide a secure means of transporting the UAS. The AUMS can transport a UAS over significant distances and through potentially hostile environments so a means of securely attaching the UAS to the AUMS is required.
- Easy integration to the host vehicle. AUMS is a fully self contained system capable of operating standalone. As such it requires no significant software modifications and minimal hardware modifications to the host vehicle.
- Minimal modifications to the air vehicle. Typically, modifications are confined to landing gear and addition of the refueling coupler. If the UAS flight control system does not provide sufficient navigation precision to repeatedly land safely on the platform, hardware and software changes to the flight control system may be required.
- Modularity. Easy to modify the system to suit host vehicle and air vehicle needs.
- Safety systems to detect and respond to fuel leakage or fire.
- Flexible fuel source and type as required. The refueling module incorporates a fuel tank or can tap into the host vehicle fuel supply as required. Compatible with gasoline or heavy fuels as needed by the air vehicle.
- Provide for partial or complete refueling as required by payload and mission considerations.

For the JCTE the AUMS host vehicle is a tele-operated High Mobility Multipurpose Wheeled Vehicle (HMMWV) UGV (Figure 2 and Figure 3). The UGV accommodates a platform diameter of seven feet without exceeding the host vehicle footprint. The 7-foot diameter platform employs eight linear actuators for centering the UAS on the platform. When the actuators are fully extended the landing surface of the platform is completely flat to eliminate the possibility of

snagging or tripping the UAS landing gear during the landing sequence. Once the UAS has landed the actuators contract toward the center of the platform and graspers erect above the platform surface to catch the landing gear and push it toward the center of the platform. The Mongoose helicopter UAS employs a 24-in diameter ring for the landing gear base. This configuration provides a capture radius of approximately 30-in. Any landing within 30-in of the center of the platform can be successfully centered with one stroke of the centering system. Landings between 30 and 38-in may be successfully captured but will require multiple strokes of the capture system. Any landing exceeding 38-in from the platform's center presents a high risk of falling off of the platform.

Figure 2. HMMWV UGV, AUMS, and the Mongoose UAS

As a matter of practice, the landing approach is continuously monitored and any approach exceeding 30-in off center is aborted. In 50+ landings to date on the platform in a wide variety of wind conditions just one abort was performed due to an intermittent altitude reading during approach. Current abort procedures are confined to automatic aborts triggered by flight control performance parameters exceeded, and manual or semi-manual aborts triggered by the UAS safety pilot or MOCU operator based on visual cues. SSC is developing a tiered automatic abort procedure which will initiate different abort responses depending on the cause of the abort. Abort parameters would include degradation of the differential global positioning system (DGPS) solution, any navigation filter status error, an intermittent loss of the above ground level (AGL) estimate from the laser range finder, or a lost link. A serious error such as a DGPS failure would completely abort the landing sequence, whereas a temporary loss of an AGL estimate may simply result in a pause in the sequence to reacquire the estimate.

Figure 3. The AUMS Landing Platform

The platform incorporates a modular core mechanism (Figure 4) that elevates and mates with the refueling coupler mounted on the bottom of the UAS once the UAS has been centered. The fluid couplers employed on both the UAS and AUMS are self sealing and spill proof. The core mechanism has a passive self-centering feature to correct for minor misalignments between the platform and UAS. The core mechanism contains the fire suppression and monitoring equipment and leak detection hardware for the platform.

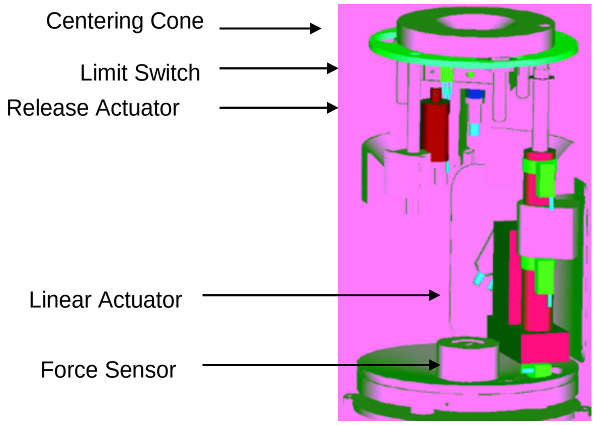


Figure 4. AUMS Platform Core Mechanism

The refueling system includes the platform core mechanism and the refueling module. The refueling module contains a 5-gallon fuel supply, bi-directional fuel pump, fuel filters, flow sensors, and fire detection/suppression and leak detection hardware. The refueling module connects to the platform via a single electrical cable with quick disconnect and a single fuel line with quick disconnect. This fuel quick disconnect automatically self seals to prevent fuel

leakage. The refueling process begins by defueling the UAS either completely empty or to a known state. Fuel is then pumped into the UAS to either partially or completely refuel the vehicle as determined by mission requirements. A fuel flow sensor is employed in the refueling system to accurately measure the amount of fuel pumped into the UAS. This approach was taken since most small UAS do not monitor actual fuel consumption or fuel quantity during operations. The refueling module is approximately 20-in x 12-in x 31-in and weighs 40 pounds without fuel. A complete refueling operation was successfully performed with the iStar UAS for the first time in late 2005.

The system control electronics are housed in an 11-in x 9-in x 9-in enclosure mounted separately from the fuel system. The enclosure contains an embedded ipEngine microcontroller, an ADR-2000 serial data acquisition board for control of fuel pump and flow control solenoids, and power conditioning. The ipEngine runs the system software to control refueling system components, actuators for centering and capture, and monitors system status. The electronics communicates with the rest of the JCTE network via a LinkSystem 802.11G router.

Command and Control of the AUMS is via the MOCU and the use of the JAUS protocol. The AUMS is a standalone system, not a payload for the HMMWV. Consequently, it is displayed in the MOCU as an independent system with its own user interface. The AUMS user interface allows the operator to center, couple, defuel, refuel and release the UAS if manual control of the process is desired. It also provides system status to monitor fuel quantity in the refueling module tank, flow rates to/from the UAS, status of the refueling process, UAS capture status, and all safety parameters for the system. Utilizing JAUS messaging for collaboration between the UAS, UGV, and AUMS the following operations are automated:

- UAS refueling service required; polls for location/status of AUMS.
- AUMS returns location and status
- UAS navigates to AUMS and executes an auto landing sequence
- UAS approach monitoring and auto abort sequence as required
- UAS sends a landed message to AUMS
- AUMS executes centering/capture/defuel/refuel operations
- AUMS indicates to UAS and operator when the UAS is refueled and ready for relaunch
- UAS or operator initiate autonomous launch

The automated sequence described above is accomplished utilizing existing behaviors and JAUS messages plus a new “landed” message to allow the UAS to inform the AUMS of its landed status. To poll for the AUMS location when required the UAS sends a standard JAUS Query Global Pose message, and the AUMS replies with a Report Global Pose message to provide its location. To execute centering, capture, defuel and refuel operations, the AUMS can use the existing Fuel Pump component and its user-defined messages. When UAS refueling is complete, the AUMS uses its Fuel Pump component’s message to indicate to the UAS and operator that UAS is ready for relaunch.

SSC worked with the iStar UAS from 2002–2006. As mentioned previously, the iStar never demonstrated sufficient precision for auto landings on the AUMS platform. Furthermore, the iStar utilized a proprietary C2 interface and was not JAUS compliant. Subsequent development

work has been conducted with helicopters beginning with a commercial-off-the-shelf (COTS) UAS helicopter, model SR-200 purchased from Rotomotion in 2005. SSC successfully demonstrated autonomous waypoint navigation with the SR-200 under the MOCU control utilizing JAUS messaging communicating through an SSC-developed translator in December of 2005. Reliability and performance issues with the SR-200 led to in-house development of new surrogate helicopters utilizing autopilots from Cloud Cap Technologies (CCT) flown first in a Caliber airframe and then in a Bergen airframe. SSC successfully demonstrated auto takeoff, auto landing, and MOCU/JAUS control of the Caliber helicopter in December 2007. However, the Caliber did not provide sufficient payload capacity to support JCTE and the Bergen suffered airframe reliability issues.

The current development airframe supporting JCTE efforts is the Mongoose UAS (Figure 5). The Mongoose is a hobby class helicopter manufactured by Airstar International. It is modified with the integration of CCT's Piccolo II autopilot, a magnetometer, laser range finder for measuring altitude above ground, and a Novatel RT-2 DGPS. A simple modification to the landing gear adds a 24-in diameter ring for AUMS centering and a refueling coupler for capture and the refueling operation. The UAS carries a simple pan–tilt gimbal with a fixed focal length electro-optical (EO) sensor to provide a limited ISR capability for demonstration purposes. The autopilot employs a neural net flight control system from Guided Systems Technologies (GST). The Mongoose UAS with the GST flight control system is capable of fully autonomous flight including auto takeoff, waypoint navigation, and auto landing. The system provides a manual pilot override for use in emergency situations. For manual pilot control a standard Futaba R/C pilot console is utilized wired into the CCT ground station. The ground station utilizes a 900-MHz band serial data radio for communication with the helicopter in both manual and autonomous flight. The helicopter autopilot incorporates lost link safeguards that force the UAS to autonomously navigate back to a predefined waypoint to attempt to re-establish communications if the link is lost for a user-defined period of time. The combination of the GST flight control system guided by the Novatel global positioning system (GPS) provides sufficient precision for repeatable auto landings within the AUMS platform capture radius. First auto takeoff, waypoint navigation, auto landing, refueling and relaunch with the Mongoose from the AUMS occurred in September 2008.

Figure 5. The Mongoose UAS

During development, determining the landing position on the platform was done by accurately surveying the launch position pre-launch and then returning to that spot for landing. The obvious

problem with this is that the UGV cannot relocate while the UAS is in flight. Both the UAS and AUMS employ highly accurate DGPS receivers. The GPS antenna position on the UAS is dictated by available space, center of gravity of the air vehicle, and electromagnetic compliance issues. On the Mongoose, the GPS antenna is located adjacent to the tail rotor at the back of the helicopter. The AUMS GPS antenna position on the HMMWV is also dictated by space available, electro-magnetic compliance, and the additional constraint of not interfering with the UAS flying in close proximity to the AUMS platform. The AUMS GPS antenna was placed on a beam attached to the forward edge of the AUMS platform in front of the HMMWV cab and below the landing surface. This results in a position offset from the ideal (landing gear of the UAS centered on the platform) which is dependent on the relative headings of the UGV and UAS. The offset is corrected using the equations shown in Figure 6:

We can measure:
ψ = HMMWV heading (deg true)
β = azimuth of platform center relative to GPS antenna
r = distance from antenna to platform
Lat_{ant} = Lat of GPS antenna
Lon_{ant} = Lon of GPS antenna

We want to calculate:
Lat_{ldg} = Lat of landing platform center
Lon_{ldg} = Lon of landing platform center

Compute earth radii assuming WGS84 and Lat_{ant}
R_m = Earth meridian radius
R_n = Earth normal radius

Compute:
$\alpha = \psi + \beta$
$Lat_{ldg} = Lat_{ant} - (r * \cos(\alpha)) / R_m$
$Lon_{ldg} = Lon_{ant} - (r * \sin(\alpha)) / (R_n * \cos(Lat_{ant}))$

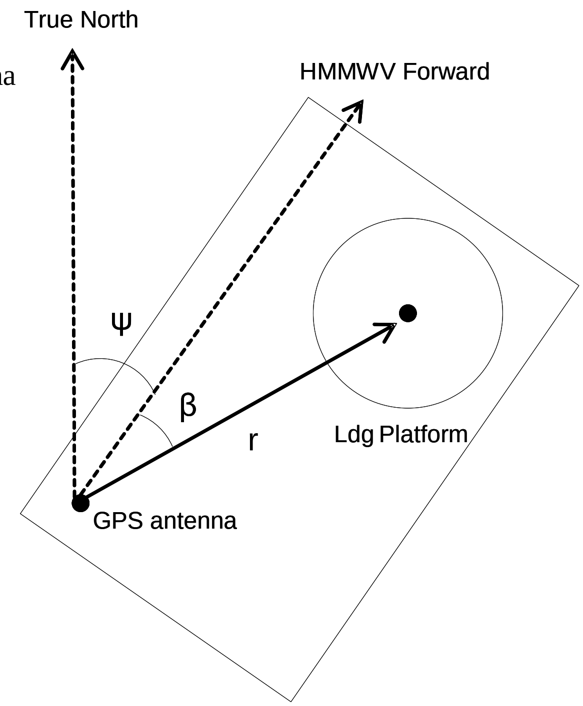


Figure 6. Offset Corrections

The CCT Piccolo II autopilot and ground station hardware used with the Mongoose UAS employ a proprietary C2 system but CCT provides a well documented application programming interface (API) for third-party developers to utilize in implementing their own C2 systems. SSC engineers developed a software translator between the API and the JAUS messages employed by the MOCU and the other JCTE components. In the system architecture employed for JCTE the CCT ground station is co-located with the AUMS platform and the HMMWV. System components such as other unmanned systems and MOCU do not communicate directly with the Mongoose UAS, they communicate via JAUS messages with the translator running at the ground station. The translator takes these JAUS messages and converts them via the API and sends them to the CCT ground station and then up to the UAS. The primary disadvantage to this configuration is that the UAS must remain within communications range of the HMMWV, approximately 10-km

with the current hardware configuration. The alternative would be to run the translator on the UAS and eliminate the CCT ground station entirely. The CCT API supports this capability, but it adds significant hardware and software development effort, increases risk, and requires additional hardware installation on the UAS with resultant increases in UAS empty weight and power consumption. The primary hardware modifications required to implement this change would be installation of an embedded micro controller running the translator software and a replacement of the Piccolo II 900-MHz serial radio with an 802.11g radio. For the first effort it was decided that risks in implementing an on-board translator on the UAS outweighed the benefits. Future plans include experimentation with this configuration.

The translator allows the MOCU to control and monitor the Mongoose UAS just like any JAUS-communicating unmanned system. Through the translator, the MOCU can command Mongoose to do automatic vertical takeoff, landing, waypoint mission and vector flying. For the vertical takeoff and landing, the translator uses custom user-defined messages to support the proprietary CCT Piccolo II API. For waypoint mission and vector flying, standard JAUS messages are sufficient. The Mongoose's position, velocity, and various running status can be translated into the corresponding standard JAUS messages and sent back to the MOCU. To minimize network bandwidth consumption, the MOCU is configured to request the translator reporting status at 1 to 2 Hz.

AUMS' role in JCTE is to demonstrate the utility of a persistent, on-demand, local airborne ISR capability at a remote location as compared to a more traditional approach utilizing fixed-wing UAS assets, which must transit to and from the home base to the remote site. The AUMS can be cued for launch in response to a threat by operators at the home base, or by other unmanned assets located on the remote site. Once airborne, the Mongoose UAS can rapidly put its EO sensor on target since there is no transit time issue. The Mongoose UAS standard fuel system provides approximately 20-min of on station time, which allows sufficient fuel reserve to return to the AUMS for refueling. Cycle time for the refueling process is approximately 4-min from the moment the Mongoose touches down until it is once again airborne. The 5-gallon AUMS fuel supply represents as much as 8 hours of flight time for the Mongoose if required. A potential enhancement to this scenario would be to utilize multiple-UAS refueling from a single AUMS, returning for refuel on a staggered schedule so at least one system is airborne at all times. The VTOL UAS also offers the potential to enhance its on-station time through perch and stare—the ability to autonomously land in an inconspicuous location and shut down while still providing ISR to the operator.

4.2.3. AUMS Host

The AUMS UAS has a 20-min flight time before it requires refueling. To extend the range of the AUMS UAS, a UGV can be used to host the AUMS refueling platform. A robotic HMMWV was designated as the AUMS host vehicle. The robotic HMMWV was selected for its payload capacity, size, and familiarity within the military community. The responsibility of the AUMS host is to carry the AUMS refueling platform to a desired location.

The robotic HMMWV required hardware, software and physical modifications to support the AUMS refueling platform. First, the AUMS refueling platform was physically interfaced to the HMMWV. Next, the HMMWV software was modified to communicate with the MOCU,

updated to the most recent version of JAUS, and implemented with teaming messages to allow greater collaboration among unmanned systems.

4.2.4. HMMWV Operation

To simplify integration efforts and to reduce risk; the AUMS host and the AUMS system were implemented completely independently (Figure 7) of each other, except for the physical interface between the two systems.

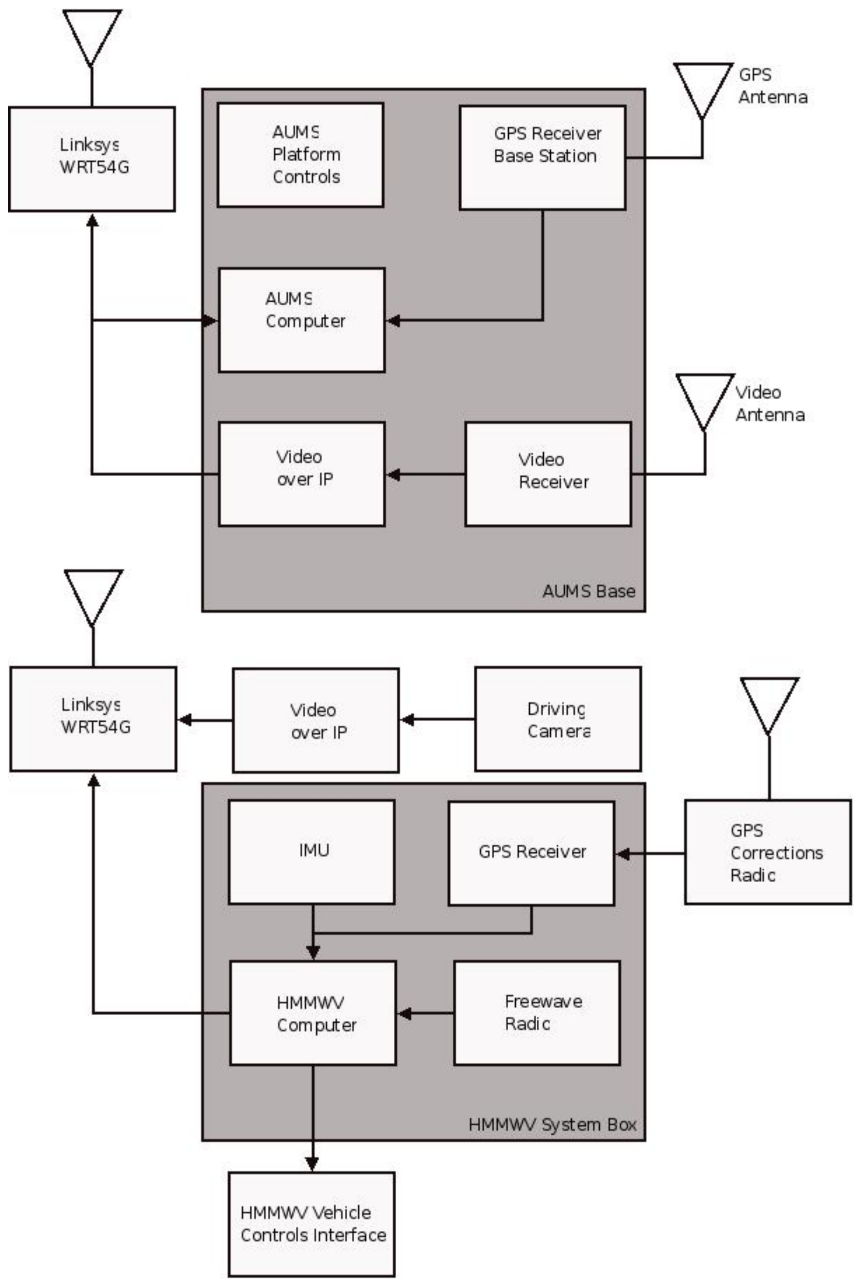


Figure 7. AUMS and AUMS Host System Diagrams

The AUMS host is a M1097A1 HMMWV, which is a high-payload configuration intended to transport equipment, material, and personnel. It has been converted to run in both tele-operation

and manual driving modes. In tele-operation mode, the AUMS host is software limited to drive 15 mph forward and 5 mph in reverse. The operator may apply the brakes, emergency stop the vehicle, and start/stop the engine. The operator may also change gears to forward, neutral, and reverse. The steering is the same Ackermann steering system as a standard HMMWV. Forward and rear driving cameras are provided on the AUMS host and the video feed is digitized using an AXIS video server with Transmission Control Protocol (TCP)-based streaming JPEGs. The video feed automatically changes from the forward to the rear driving camera when the operator changes the gear from forward to reverse and vice versa. The robotic HMMWV was used previously for the CEE effort.

In addition to providing basic tele-operation, the HMMWV contains a 3DM-GX1 Microstrain inertial measurement unit (IMU), which senses the roll, pitch, and yaw of the vehicle and reports it to the HMMWV computer. It has an orientation resolution of < 0.1° and ± 2°-in dynamic conditions. Also, a Novatel GPS receiver with 10-meter uncorrected accuracy reports the HMMWV GPS location. To provide more accurate readings, a serial radio is located on the HMMWV to receive differential corrections for the GPS receiver. However, for this demonstration the corrections receiver was not used.

For this demonstration, the robotic HMMWV was controlled through the MOCU. All communications with the MOCU were accomplished through an ESTEEM Ethernet radio, which talked to the another ESTEEM radio on the Yamaha RMAX rotary wing UAS as shown in Figure 8. All settings such as frequencies, ports used, etc., may be found in the Interface Design Document (IDD).

Figure 8. Yamaha RMAX

4.2.4.1. HMMWV Specifications

Model	M1097A1
Engine/Power	8 Cyl, 6.5 Liter
Rating	150 hp @ 3600 rpm
Width	85-in
Height	69-in (w/o AUMS platform)
Ground Clearance	16-in Under Axle 24-in Under Chassis
Length	180-in
Weight	5600-lbs

Vehicle Curb Weight	10000-lbs
Max Payload Capacity	4400-lbs
Range	275-mi

4.2.5. AUMS Integration

4.2.5.1. Description

For the AUMS to accomplish its refueling mission, a sufficiently strong physical interface between the AUMS refueling platform and the robotic HMMWV was required to withstand the extra stress and vibrations associated with driving. Additionally, the platform had to be properly positioned to minimize the risk of collision with the robotic HMMWV during takeoffs and landings. Large protruding components, such as antennas, represented the greatest risk for collisions.

4.2.5.2. Integration

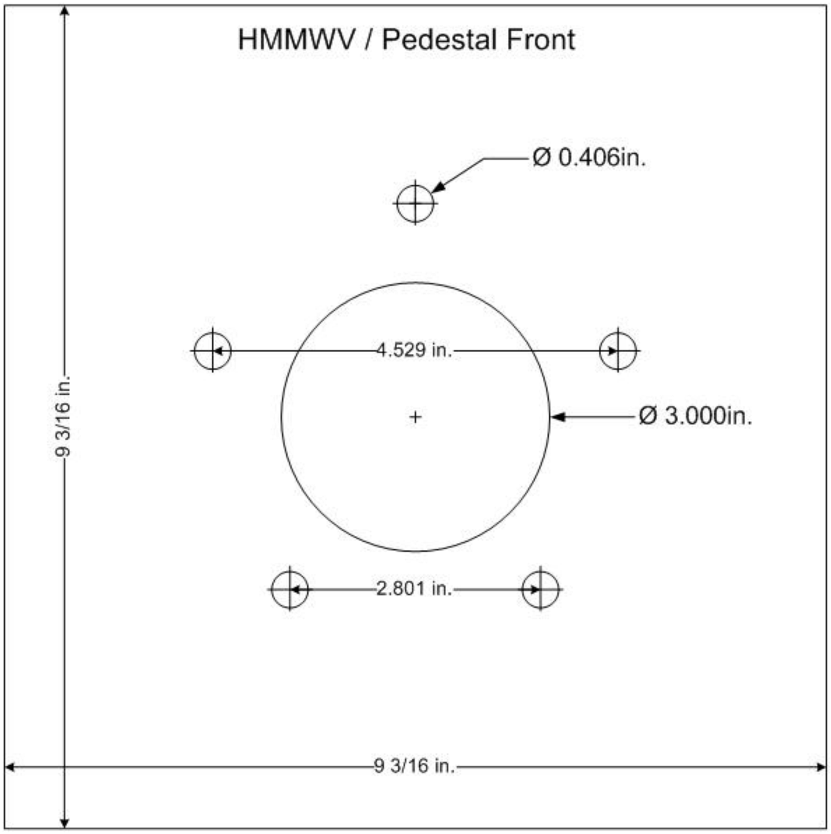
The AUMS refueling platform was placed on a pedestal mount on the robotic HMMWV. This created the highest and sturdiest position available on the vehicle. The pedestal was originally designed as a turret mount to support a TeleRobotics Corporation Remotely Operated Weapons System (TRC XROWS) turret for a previous, unrelated effort (Figure 9). To place the AUMS on the pedestal, the turret was removed from the pedestal and the mounting pattern was photographed and documented as shown in Figure 10 and Figure 11. It was determined that the AUMS refueling platform could be modified and integrated on the pedestal, and that the interface to the turret mount would be sufficiently strong to support both the refueling platform and the UAS.

Figure 9. Turret Mount with Turret Attached

Figure 10. AUMS Host Turret Mounting Pattern

4.2.5.3. Results/Outcomes

The AUMS refueling platform was attached to the turret base on the HMMWV according to plan. The fuel cell was fastened to the pedestal support bars and the AUMS electronic components were placed underneath the turret shown in Figure 12. To reduce the risk of the UAS colliding with antennas on the AUMS host; all antennas used by the AUMS host were moved to the corners of the frame of the HMMWV. This configuration worked well for the duration of the JCTE demonstration at Tyndall Air Force Base.



½ Scale

Figure 11. Turret Base Diagram

Figure 12. AUMS Host with AUMS Platform Attached

4.2.6. MOCU Compatibility

4.2.6.1. Description

To accomplish its mission, the AUMS host establishes a communication link with MOCU. The communication link is established through the RMAX communications repeater and directional antenna. To achieve MOCU compatibility, a portion of the command class and core subgroup of the JAUS/AS-4 message set was integrated into the robotic HMMWV system. The implemented messages include: Request Component Control, Release Component Control, Confirm Component Control, and Reject Component Control. This task was more challenging than originally expected because the HMMWV was initially designed to communicate with only a single pre-specified controller and not in a network environment. The majority of problems occurred when MOCU assumed control of the HMMWV and subsequently released control.

4.2.6.2. Integration

The AUMS host was successfully operated by the MOCU during integration activities leading up to the demonstration. In this environment the HMMWV was communicating through the ESTEEM radio to a local antenna, to which MOCU had access. However, when the other unmanned vehicles were added to the network in the same environment the AUMS' host signal was detectable but not reliable. This issue was attributed to the antenna location at the time. After the demonstration activities had ended, troubleshooting in greater depth indicated the likely causes to be in the AUMS host software or due to interference problems with the ESTEEM radios.

4.2.6.3. Teaming Messages

Teaming messages were meant to enhance the collaboration capabilities of a team of robots. They allow a robot who has been given a specified mission, i.e., the Team Leader, to query other robots within communications range. It sends out a broadcast to find robots with capabilities that may help accomplish the mission. When the Team Leader receives the neighboring robots' capabilities, the Team Leader or an operator may choose which robots to allow on the Team. Once they have joined a Team these robots may utilize their available capabilities to assist in accomplishing the Team Leader's mission. Once the mission is accomplished, the Team is disbanded and they wait for another mission to be requested.

4.2.6.3.1. Implementation

JAUS Teaming Message List

- Code DA00h: Request Team Leadership/Membership
- Code DA01h: Reply Team Leadership/Membership
- Code DA02h: Release Team Membership
- Code DA03h: Add Team Member
- Code DA04h: Remove Team Member
- Code EA05h: Query Team Membership
- Code FA05h: Report Team Membership
- Code DA06h: Request Peer Connection
- Code DA07h: Set Peer Connection
- Code DA08h: Terminate Peer Connection

4.2.6.3.2. Message Descriptions

Code DA00h: Request Team Leadership/Membership: The DA00h message (Request Team Leadership/Membership) is used to request that a component join a Team either as the Leader or as a Member. If the component can support Team Lead/Member functionality, it can then assume Team Lead or Team Membership or report that Team Lead/Member functionality is not supported. If Team Lead/Member functionality is supported, the receiving component must compare the Team Lead ID in the message with its own Source ID. If the IDs match, it should recognize that the request is for it to assume leadership of a Team. If the IDs do not match, then it should recognize that the message is a request for it to join a Team. Upon establishment as a Team Leader, the receiving component will be able to create Teams to control directly or to pass messages to/from a higher authority. The authority code provided by the requestor is assumed to be that of its direct superior. The component therefore assumes an authority code of one less than the originator. When a component joins a team, it takes note of the authority of the requesting component. If another Team Membership request is received, the authority in the message is compared to the original authority. If the new authority is higher, the component joins the new team. If the authority is equal to or lower than the one in memory, membership is not accepted. Figure 13 shows the Team Leadership/Membership messaging decision process flow. The team designation is the Team Lead's source ID (Table 1).

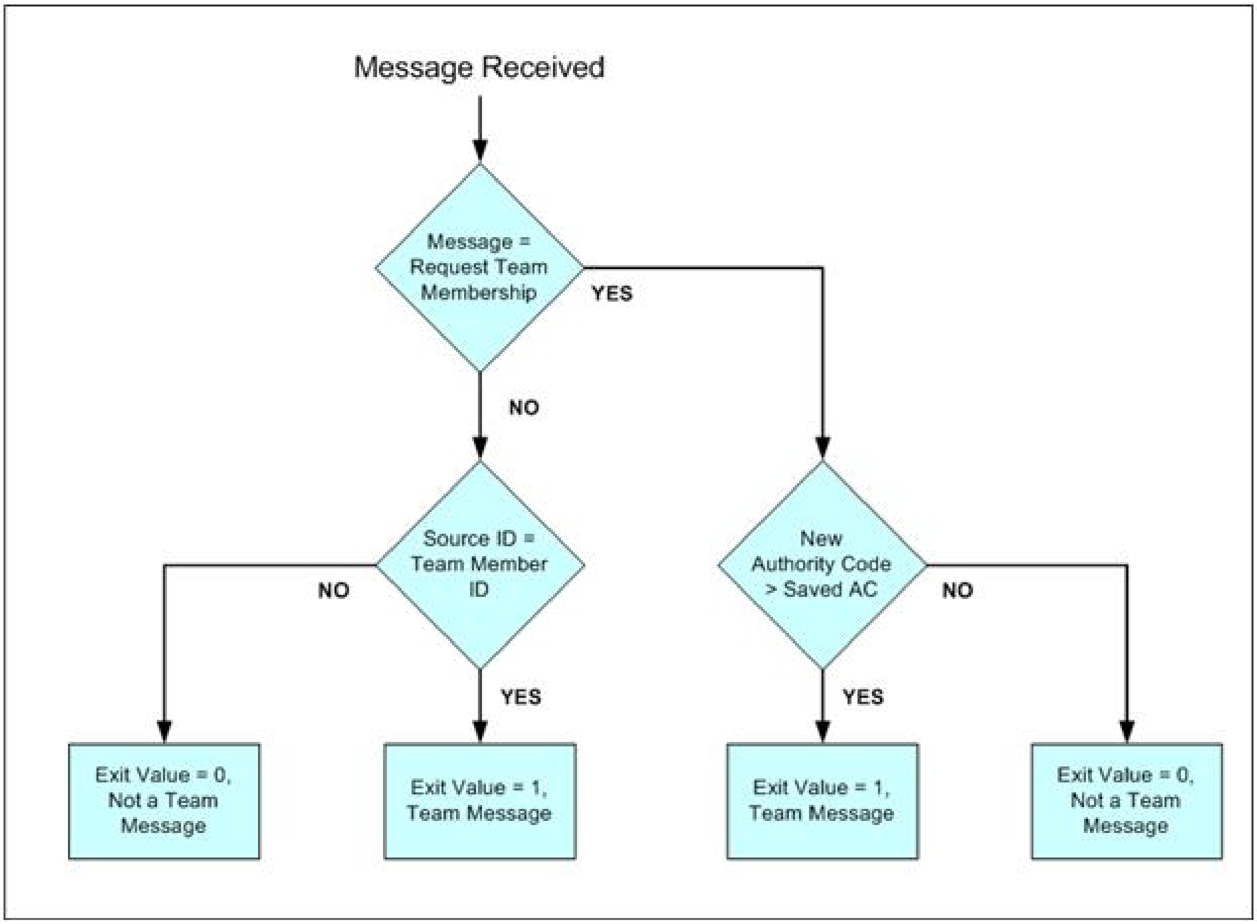


Figure 13. Decision Tree for Evaluating Team/Teaming Messages

Table 1. JAUS Byte Field Population for Message DA00h: Request Team Leadership/Membership

Field #	Name	Type	Units	Interpretation
1	Authority Code	Byte	N/A	Authority 0-255
2	Team Lead	4-Bytes	N/A	Source ID of Team Leader

Code DA01h: Reply Team Leadership/Membership: The DA01h message (Reply Team Leadership/Membership) is used to accept or reject a Team Leadership/Membership request from the requesting component. When “Team Leadership” or “Team Membership” is accepted, with a response code of “0,” the component will then be able to establish or join a Team. It then generates, passes or accepts team messages. Additionally, it will choose to allow or deny peer connections between its subordinate team members (if any) and outside requestors.

If the component has already established a Team of its own, it should not receive another Team Leadership request. If this is the case, the message likely originated from another component with a lower authority than its Team Lead. Any such requests would be responded to with a code of “1,” “Leadership Not Accepted.” For components not supporting Team Leadership control capability, the response code value of “2” shall be used.

If the component does not belong to a Team, or already belongs to a Team and a Team Membership request arrives from an authority higher than its Team Lead’s, it will then join the new Team and respond with a response code of “0.” If the component belongs to a Team and a Team Membership request arrives from an authority equal to or lower than its Team Lead’s, it will respond with a response code of “1,” Membership Not Accepted. For components not supporting the Team Leadership control capability, the response code value of “2” shall be used (Table 2).

Table 2. JAUS Byte Field Population for Message DA01h: Reply Team Leadership/Membership

Field #	Name	Type	Units	Interpretation
1	Response Code	Byte	N/A	Bits 0 and 1: 0 = Leadership\Membership accepted 1 = Leadership\Membership not accepted 2 = Leadership\Membership not supported Bits 3–7: Reserved

Code DA02h: Release Team Membership: The DA02h message (Release Team Membership) is used to relinquish team membership of the receiving component. This command is accepted only if received from the Team Leader or from a component of higher authority than the one which sent the original Team Membership message (Table 3).

Table 3. JAUS Byte Field Population for Message DA02h: Release Team Membership

Field #	Name	Type	Units	Interpretation
1	Authority Code	Byte	N/A	Authority 0-255

Code DA03h: Add Team Member: The DA03h message (Add Team Member) is used once a component has accepted membership within a Team. Other Team Members may be made known to the component using this command. This command is sent to the component from the Team Lead. The component is responsible for holding a list of Members within its Team (Table 4).

Table 4. JAUS Byte Field Population for Message DA03h: Add Team Member

Field #	Name	Type	Units	Interpretation
1	Team Member	4-Bytes	N/A	Source ID of new Team Member

Code DA04h: Remove Team Member: The DA04h message (Remove Team Member) is used when components are reassigned to other Teams. This message is used to inform the Members of the remaining Team that the Member has left the Team. This command is sent to the component from the Team Lead. The component is responsible for removing this address from the list of members within its Team (Table 5).

Table 5. JAUS Byte Field Population for Message DA04h: Remove Team Member

Field #	Name	Type	Units	Interpretation
1	Team Member	4-Bytes	N/A	Source ID of new Team Member

Code EA05h: Query Team Membership: The EA05h message (Query Team Membership) is sent to a component to inquire what Team it is assigned to.

Code FA05h: Report Team Membership: The FA05h message (Report Team Membership) is a response to the Team Membership query. It serves to inform the requestor of the designation assigned to that component's Team and Team Leader (Table 6).

Table 6. JAUS Byte Field Population for Message FA05h: Report Team Membership

Field #	Name	Type	Units	Interpretation
1	Team Leader	4-Bytes	N/A	Source ID of Team Leader

Code DA06h: Request Peer Connection: The DA06h message (Request Peer Connection) is used to request a peer connection between the receiving component and a sending component. This message is sent to the component's Team Leader if one exists. If the Leader does exist, this request is accepted or rejected by that component's Team Lead. When established, the receiving

component shall execute commands only from the Team Lead or peer until the connection is terminated (Table 7).

Table 7. JAUS Byte Field Population for Message DA06h: Request Peer Connection

Field #	Name	Type	Units	Interpretation
1	Team Member	4-Bytes	N/A	Source ID of team member desired

Code DA07h: Set Peer Connection: The DA07h message (Set Peer Connection) is used to set a peer connection between the receiving component and a sending component. Typically, this message is sent by the Team Lead to both the requesting component and the subordinate Team Member, informing both of the grant status. If the Team Lead denies the request for a peer connection, then only the requestor receives this message, with a connection code of “0.” Otherwise, both receiving components receive a message with a sequentially numbered connection code that both associate with the specific peer connection granted. The source ID sent to each component is the source ID of the peer which it will establish a link with (Table 8).

Table 8. JAUS Byte Field Population for Message DA07h: Set Peer Connection

Field #	Name	Type	Units	Interpretation
1	Authority Code	Byte	N/A	Authority 0-255
2	Connection Code	Byte	N/A	Connection 0-255
3	Team Member	4-Bytes	N/A	Source ID of Peer

Code DA08h: Terminate Peer Connection: The DA08h message (Terminate Peer Connection**)** is used to terminate a connection that has been established between two components using a peer connection (Table 9).

Table 9. JAUS Byte Field Population for Message DA08h: Terminate Peer Connection

Field #	Name	Type	Units	Interpretation
1	Authority Code	Byte	N/A	Authority 0-255
2	Connection Code	Byte	N/A	Connection 0-255

4.2.7. Results

Teaming messages would greatly enhance the collaborative capabilities of unmanned assets if they had the ability to solve problems on their own or accomplish a mission or task on their own. However, current unmanned assets are far from achieving these capabilities. Therefore these teaming messages were not implemented on all of the systems in the JCTE demonstration.

4.2.8. Interim Findings/ Lessons Learned

The AUMS host was able to communicate with the MOCU on a one-to-one basis without any problems, but the HMMWV had difficulties maintaining a signal when communicating through

the S-band and L-band links. The HMMWV message handling software was intended to work singularly with one controller, not in a network environment. Because of this, the video and HMMWV control signals were intermittent or unavailable when all of the unmanned assets were connected to the network. A solution was quickly implemented to improve communications with the AUMS host. The solution was to reposition the radios located on the HMMWV. The GPS antenna, the ESTEEM radio, and the AXIS video server were mounted on the corners of the vehicle and raised as high as possible without interfering with the AUMS UAS's flight space. This added some improvement to signal quality and strength but the signal was still intermittent. More testing is needed to pinpoint the cause of this problem. Some upgrades are also needed to the AUMS host communications code to make it more robust in a network environment.

Another issue that surfaced during the integration activities and the demonstration was that the AUMS host computer would completely shut down if left on the network for more than 30 minutes. This was attributed to the fact that the HMMWV was not only trying to communicate with the MOCU, but it was also trying to communicate with some of the other unmanned vehicles on the network. It was receiving more messages than it could handle, causing a stack overrun, which caused the system to stop responding completely. The quick fix to this problem was to manually reboot the system by using the emergency stop mechanism when the system shut down. This problem needs to be addressed by implementing software fixes in the message handling code of the AUMS host.

The video stream was unreliable. This may be due to sending the video using TCP protocol. TCP protocol by its nature is not well-suited to a lossy environment. Lossy compression is a data encoding method that compresses data by discarding (losing) some of it. If a packet is lost, the receiver must respond to the transmitter and tell it to resend the packet. In the meantime, the receiver is stalling and waiting for the packet to be resent. This causes more transmission delay than usual, resulting in an intermittent video signal when the probability of a lost packet is high when the Axis Video Server is communicating through the S-band and L-band. A possible solution is to use User Data Protocol (UDP) for video feeds.

The teaming messages were implemented on the AUMS host, but because of time and budget constraints they were not implemented on other systems. For that reason, the teaming messages were not able to be fully tested nor demonstrated. Further testing and demonstration of this capability remains a possibility in the future.

4.2.9. Future Improvements

The weakest link of the AUMS host system is the communications subsystem. This includes the software and the radio and antenna setup. The changes that were made to update the software to the JAUS/AS-4 message set need to be reviewed and improved. Additionally, the video feed should be changed to utilize the UDP protocol instead of the TCP protocol.

4.3. Unmanned System (UMS) Communication Repeater (UCR)

4.3.1. Technical Description

The UMS Communication Repeater (UCR) is a bi-directional radio frequency (RF) digital data repeater developed to support BLOS networked communication between one or more operators and one or more UMS. The UCR extends the effective range of operation for a UMS

communication network based on 802.11 Wireless Fidelity (WiFi) to distances that are beyond visual range. This is accomplished by placing a communication repeater node in the air on a UAS as a self-contained payload.

As shown in Figure 14 the UCR provides an airborne link between an OCU and one or more UGVs. The system is represented in this figure by the two yellow boxes comprised of the tracking antenna controller and the Comm Repeater Ground Unit. The overall link is actually implemented via two separate links, an L-band between the OCU and UAS, and an S-band between the UAS Comm-Payload UGVs (Figure 15). The L-band link is implemented as a Frequency Modulation (FM) telemetry uplink/downlink operating on two separate frequencies. The S-band link is essentially WiFi conforming to 802.11 b/g.

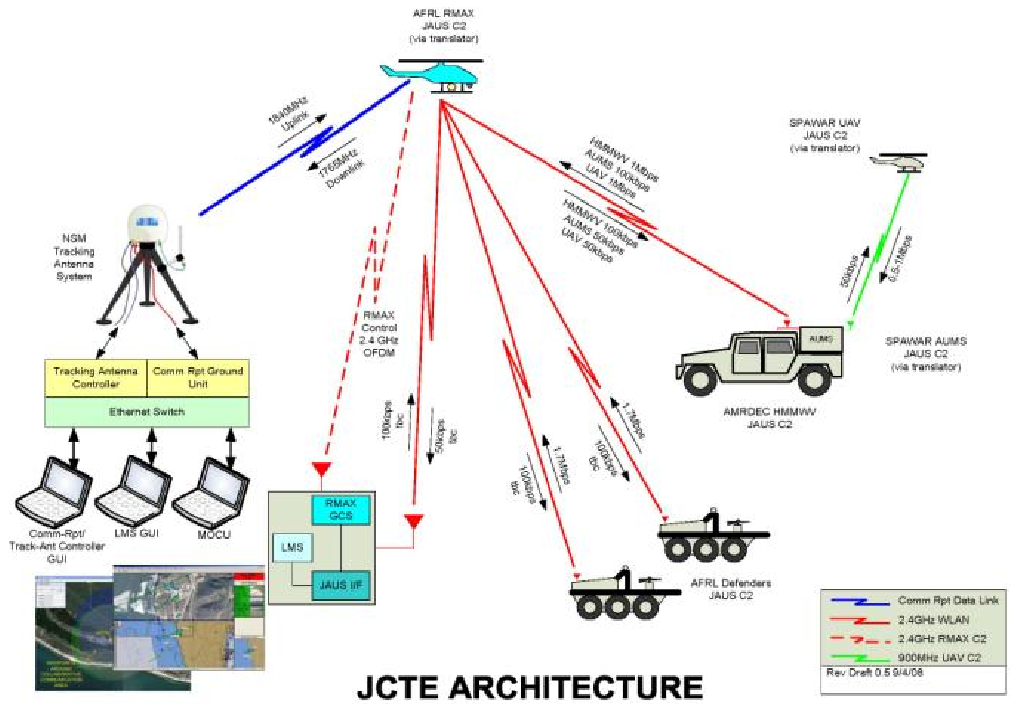


Figure 14. JCTE Communication Scheme

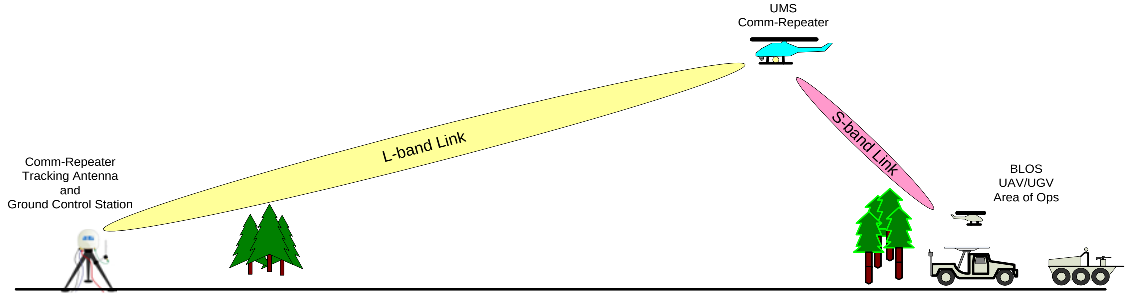


Figure 15. L-Band & S-Band Link

The theory of operation is as follows. The OCU interfaces to the Comm-Package through a standard CAT-5 type hard wired Ethernet interface. Operator commands typically sent to a selected UGV via a WiFi network are picked-up by the OCU Comm-Package. The Ethernet data packets destined for an UGV (i.e., UGV Internet Protocol (IP) address) on the opposite side of the link are picked up by a single-board computer (SBC) located in the Comm-Package. Essential data and transport information are stripped out of the Ethernet packet and packaged into a high-level data link control (HDLC) frame. The HDLC frame is subsequently transmitted to the Comm-Repeater via the L-band link. A second SBC located in the Comm-Repeater pulls the payload data out of the received HDLC frames and retransmits this information to the intended recipients via an 802.11 WiFi access point (AP) across one or more S-band links established between the AP and UGVs. Status and video data flows from the UGV(s) to OCU in a similar manner.

System specifications for the UCR are as follows:

OCU Comm-Package
19-in Rack Mount Flexi-Box, 3.5-in front panel height
Front Panel Control/Monitor with Rear Panel Input/Output (I/O)
Size: 3.5-in x 19-in x 14.5-in (Height x Width x Depth)
Weight: 25-lbs
Interfaces:
Power: 115 VAC/60 Hz
Data: RJ-45 Ether Data I/F to OCU network
Frequencies:
L-band Uplink – 2840-MHz
L-band Downlink – 2765-MHz
Transmitter (Tx) Power:
L-band, 2 Watt/10 Watt selectable

Comm-Payload

Hardware mounted to 24-in x 7-in x 0.25-in (Length x Width x Thick) Aluminum (AL) plate
Designed for dual carriage on fixed- or rotary-wing UAS:
Plate mounted within internal payload bay
Or mounted inside 7.5-in inner diameter pod for external carriage

- Power: 28 VDC @ 7.5 Amps max
- Volume 5.5-in x 7-in x 24-in (Height x Width x Depth)
- Weight:
    - 15-lb (w/o Pod)
    - 25-lb w/ 7.5-in inner diameter x 24-in long Pod

Frequencies:
    - L-band Uplink – 2840-MHz
    - L-band Downlink – 2765-MHz
    - S-band – 2400 to 2475-MHz
- Tx Power
    - L-band, 2 Watt/10 Watt selectable
    - S-band, 1 Watt (or lower; selectable)

4.3.2. JCTE Technical Objectives

The UCR capability was originally developed and demonstrated with the Comm-Payload carried internal to a small fixed-wing UAS in 2003. For the JCTE effort this capability was modified to provide certain performance improvements as well as for test and demonstration on a rotary-wing UAS. Technical objectives established for the UCR under the JCTE effort included the following:

- Increase Data Throughput from 1 Mbps to 6 Mbps
- Increase Effective Range at and Beyond 20-mi
- Integrate/Test Comm-Payload on a Rotary-Wing Platform
- Reduce Operator Workload

These objectives were met in several different ways. The increase in data throughput was achieved through hardware and software modifications. A Bit-Sync card and higher-speed serial commutation controller (SCC) was added to both the Comm-Package and Comm-Payload. With the new cards in place, software was modified to effect changes in clock speeds and thus data rates. These changes were tested at the bench as RF-over-cable as well as in free-space RF transmission. Bench level testing up to 8 Mbps was achieved; however, this data rate was backed down to 6 Mbps to increase the reliability of data transmission in free space to add fade margin that accounted for dynamic changes in vehicle attitudes and thus signal strengths.

The increase in effective range was achieved through the addition of a high-gain dish tracking antenna system to the ground control station side of the system. This tracking antenna system was a commercial item procured from NS Microwave and provided a 29 dB gain, 17 dB more gain than previously achieved through use of a sector antenna, and 24 dB more gain than the omni-stick antenna that was being used for close-in UAS operation.

The Comm-Payload was integrated to a Yamaha RMAX rotary wing UAS. To accomplish this, the original Comm-Payload equipment that was mounted to a 24-in x7-in AL plate was installed in a 24-in long 7.5-in diameter tube. This tube was fabricated out of spun fiberglass and utilized AL end plates to which the Comm-Payload was affixed. Two attaching points suspended the tube from underneath the belly of the RMAX UAS. To make the Comm-Payload a self-contained package, LiIon battery packs were procured and mounted in an external box that was affixed to the Comm-Payload tube. A single LiIon battery pack was able to transmit power in the L-band Link for up to 60-min at 10W and 90-min at 2W.

To accomplish the last technical objective—reduce operator workload—the UCR was designed for minimum operator setup and ease of operation. Following initial power-up and tracking antenna system alignment, the UCR operates transparently to the unmanned vehicle operators. Link status can be monitored and recorded via a software application running independent from the OCU(s). No other operator intervention is required.

4.3.3. JCTE Integration Effort

The JCTE integration effort related to the UCR was focused in two areas:

- Integrating the Comm-Payload to the RMAX UAS
- Integrating the Tracking Antenna System to the JAUS Backbone

As previously described, the Comm-Payload was installed into a 24-in long by 7.5-in inner diameter fiberglass tube. This integration included modifying two 7.5-in diameter aluminum endplates to which the Comm-Payload plate, and electronics, was attached. One end plate was modified to contain operational controls (switches, indicators) and connectors for power and data interfaces. A separate battery box was fabricated out of aluminum and affixed to the Comm-Payload. This box housed a single 28 VDC LiIon battery pack, which interfaced to the I/O panel of the Comm-Payload via a power cord. The Comm-Payload was easily suspended from the center mid-body of the RMAX UAS by two shock-isolated mounting brackets. The two antennas, L-band blade and S-band blade, were initially mounted to the pod. Later, these antennas were moved to other locations to minimize signal shading by the UAS.

In a separate effort, the tracking antenna procured from NS Microwave was integrated to the Comm-Package and JAUS backbone. L-band RF components previously contained in the Comm-Package enclosure were relocated by NS Microwave to the RF pedestal to minimize RF signal loss over cable. This resulted in changes to the original Comm-Package enclosure and I/O interfaces. A serial data interface was added to the back of the box for interfacing to the controller electronics in the tracking antenna head. Also, a software module was developed to send UAS position reports from the OCU to the tracking antenna controller. Thus, the tracking antenna controller was interconnected to the JAUS backbone.

4.3.4. Test and Evaluation

Test and evaluation of the UCR was performed in a stage-wise manner. Data bandwidth improvements were first tested in the lab over wire without RF components. RF components were then introduced and testing was performed over coax cabling using RF attenuators to simulate path loss. At this stage testing was performed using certain message error rate tools and unmanned system communication simulators. This eventually led up to field testing, where system performance was evaluated with free-space RF transmissions and representative hardware that included OCU and UGV equipment.

Field testing was also done in a stage-wise manner. Testing of the UCR was first performed with the Comm-Payload in a fixed location while using RF attenuators to control radiated signal strength. In this manner the system could be evaluated without the effects of UAS vehicle dynamics. Equipment was in close proximity during this stage of test. After satisfactory results were obtained, the Comm-Payload was mounted on the RMAX UAS and flown at low altitudes above the test range with the OCU and track antenna within 1 mile of the UAS and ground vehicle area of operations.

Following close-in range testing the OCU and Comm-Package (to include tracking antenna) were moved to fixed distances of 5, 10, 12 and 15 miles. At each OCU test site the RMAX UAS was put in the air and communication was established between the operator(s) located at the OCU and one or more UGVs. Vehicle operations performed via the UCR were assessed along with link stability. Test sites to evaluate the UCR at longer ranges were identified out to 20-mi but coordination of these test sites, as well as the UAS flight altitudes required to support these tests, were not obtained during the period of this effort.

Integration testing led up to the JCTE Demonstration, during which the UCR supported BLOS unmanned vehicle operations at a distance of 5-mi. Results of these integration and test activities are described in the following section.

4.3.5. Results

Results of UCR integration, test, and evaluation are as follows:

- UCR tested out to 15-mi
- Data throughput sustained at 6 Mbps
- Supported multiple operators working from multiple OCUs
- Maintained BLOS operations of multiple UGVs
- UCR L-band link performance greatly improved by Tracking Antenna System; need to pay attention to Fresnel Zone limitations during setup/configuration.
- UCR S-band link performance still needs to be refined.

4.3.6. Recommendations

Based on lessons learned during JCTE experimentation and demonstration the following recommendations have been made to improve the performance of the UCR:

- Improve performance of S-band link(s)
- Automate setup/configuration of tracking antenna system
- Test at extended ranges (out to 50-mi)

S-band link performance is greatly affected by vehicle dynamics and antenna types. Typically omni-directional antennas are used on the ground vehicles. These antennas have toroidal patterns that are optimum in the horizontal plane but roll off significantly with increase in elevation. When using an airborne employed communication repeater node such as the UCR, the air vehicle might be at a relatively high-elevation (aka look-up) angle with respect to one or more ground vehicles. Typically the higher the look-up angle the lower the signal strength. In addition, the air vehicle will be changing in attitude, which also attributes to scintillation in signal strength. These factors can be mitigated through antenna optimization. Directional or beam steering would contribute greatly to an increase in S-band link performance while adding some additional system complexity. These technologies and their applicability to UMS communication networks warrant further investigation.

The effectiveness of the tracking antenna system used with the UCR is highly dependent on accurate North alignment, position location, and leveling. At present this setup and configuration is done manually. It is believed that this setup can be automated to reduce the overall operator workload associated with utilization of this equipment.

Theoretically the UCR with tracking antenna system should support BLOS UMS operations out to 50-mi. Range and terrain limitations along the Gulf Coast inhibited additional testing of the UCR at ranges longer than 15-mi. To test at extended ranges the line-of-sight (LOS) between the OCU equipment and thus tracking antenna must be obtained. This will require two things: low-elevation obstruction along the bearing from the tracking antenna to UAS, and UAS operation at higher altitudes. The first can be achieved through proper site selection. The second item, higher UAS operating altitudes, will need to be coordinated with the Federal Aviation Administration (FAA) and Eglin/Tyndall Air Traffic Control (ATC) and flight operations.

4.4. Link Management System (LMS)

4.4.1. Technical Description

In general, the Link Management System or LMS is a software module developed to automate management of dynamically created mobile and fixed wireless networks supporting UMS operations. The LMS uses *a priori* knowledge of performance parameters associated with all participants, fixed and mobile, that may join the network. This knowledge along with the dynamically updated position information for each participant is used to compute the region of effective communication for each transmitting/receiving node. Using well defined and tested RF path loss algorithms, the LMS computes the effective communication region for each participant in the network. Overlapping coverage areas identified by the LMS represent regions in which one or more participants are able to communicate with each other. The output of the LMS can be used for dynamic path planning and intermittent or lost communication response management.

During the JCTE effort the LMS was effectively used to determine the optimum location for placement of the UAS carrying the UCR Comm-Payload. The LMS determined the effective communication region for the L-band link between the OCU and UAS carrying the Comm-Payload, and again for the S-band link between the UAS and ground vehicles. Communication equipment parameters (i.e., Transmiter (Tx) power, Receiver (Rx) sensitivity, antenna gains, etc.) for each participant in the network were loaded into a configuration file that was read by the LMS upon startup. Participant position reports were constantly monitored by the LMS, which was connected to the JAUS backbone. Based on these inputs the LMS computed the optimum location for the RMAX UAS to establish and sustain communications between the OCU and UGVs. This position was reported across the JAUS network to the RMAX controller as a fly-to waypoint. When enabled the RMAX would automatically fly to the optimum position reported by the LMS. This capability was tested, evaluated, and demonstrated during the JCTE effort.

A separate software application called the LMS GUI was developed to provide a single Common Operating Picture (COP) or visual interface of vehicle positions and LMS status to an operator. The LMS GUI provided a map base of the area of operations with overlaid graphics that depicted OCU and vehicle positions as well as areas of effective communications. The LMS GUI also graphically depicted the optimum location or waypoint to which the RMAX UAS should be located at any given instant in time based on reported positions of the OCU and all ground and air vehicles. The LMS GUI application could be operated on any computer connected to the JAUS backbone.

4.4.2. Algorithm Definition

4.4.2.1. Positional Data

Input positional data for the OCU, UAS, and UGV(s) is entered into the system in GPS coordinates (Latitude, Longitude and Elevation) in a World Geodetic System (WGS) 84 format [3]. To generate the user display the positions are converted to local East–North–Up (ENU) coordinates referenced to the OCU position.

The conversion is a two step process:

1. GPS to Earth Centered Earth Fixed (ECEF) coordinates
2. ECEF to ENU coordinates

The reverse conversion process will be done to generate the optimum waypoint for the UAS. This conversion is also in two steps:

1. ENU to ECEF coordinates
2. ECEF to Geodetic Coordinates

4.4.2.1.1. GPS to ECEF Conversion

1. $VEH_X = \left(\frac{a}{\chi} + h\right)\cos\phi\cos\lambda$
2. $VEH_y = \left(\frac{a}{\chi} + h\right)\cos\phi\sin\lambda$
3. $VEH_z = \left(\frac{a(1-e^2)}{\chi} + h\right)sin\phi$
4. $OCU_X = \left(\frac{a}{\chi} + h\right)\cos\phi\cos\lambda$
5. $OCU_y = \left(\frac{a}{\chi} + h\right)\cos\phi\sin\lambda$
6. $OCU_z = \left(\frac{a(1-e^2)}{\chi} + h\right)sin\phi$

Where:

a = 6378137.0; earth semi-major axis in meters
$e^2 = 6.6943799014 \times 10^{-3}$; first eccentricity squared value
$\chi = \sqrt{1-(e^2\ \sin^2\phi)}$; $\frac{a}{\chi}$ is the normal distance from the surface along the Z-axis
h = elevation
$VEH_{x,y,z}$ = X,Y,Z ECEF coordinates of an arbitrary unmanned vehicle system (i.e. *UAS,UGV, or USV)*
$OCU_{x,y,z}$ = X,Y,Z ECEF coordinates of the OCU
ϕ, λ = the latitude and longitude respectively of the defined variables VEH & OCU

4.4.2.1.2. ECEF to ENU Conversion

1. $VEH_e = -\sin\lambda\,(VEH_x - OCU_x) + \cos\lambda\,(VEH_y - OCU_y)$
2. $VEH_n = -\sin\phi\cos\lambda\,(VEH_x - OCU_x) - \sin\phi\sin\lambda\left(VEH_y - OCU_y\right) + \cos\phi\,(VEH_z - OCU_z)$
3. $VEH_u = \cos\phi\cos\lambda\,(VEH_x - OCU_x) + \cos\phi\sin\lambda\left(VEH_y - OCU_y\right) + \sin\phi\,(VEH_z - OCU_z)$

Where:

$VEH_{e,n,u}$ = ENU coordinates of an unmanned vehicle system with respect to the OCU
ϕ = the latitude of the OCU
λ = the longitude of the OCU

4.4.2.1.3. ENU to ECEF

1. $X = -\sin\lambda\,VEH_e - \sin\phi\cos\lambda\,VEH_n + \cos\phi\cos\lambda\,VEH_u + OCU_x$

2. $Y = \cos\lambda\, VEH_e - \sin\phi \sin\lambda\, VEH_n + \cos\phi \sin\lambda\, VEH_u + OCU_y$
3. $Z = \cos\phi\, VEH_n + \sin\phi\, VEH_u + OCU_z$

Where:

$X = VEH_x$
$Y = VEH_y$
$Z = VEH_z$
$VEH_{x,y,z}$ = ECEF coordinates of an unmanned vehicle with respect to the OCU
ϕ = the latitude of the OCU
λ = the longitude of the OCU

4.4.2.1.4. ECEF to Geodetic Coordinates

1. Latitude $\Phi = a\tan\left(\frac{Z + \left(e^{12} * b * \sin^3(\theta)\right)}{p - \left(e^2 * a * \cos^3(\theta)\right)}\right)$
2. Longitude $\lambda = a\tan\left[\frac{Y}{X}\right]$
3. Height = $\frac{p}{\cos(\Phi)} - N$

Where:

a = 6378137.0; earth semi-major axis in meters
b = 6356752.3142, earth semi-minor axis in meters
e^2 = 6.69437999014x10^{-3}; first eccentricity squared

$$e^{12} = \frac{a^2 - b^2}{b^2}$$

$$p = \sqrt{X^2 + Y^2}$$

$$\theta = a\tan\left(\frac{Za}{pb}\right)$$

$$N = \frac{a}{\sqrt{\left(1 - \left(e^2 * \sin^2(\Phi)\right)\right)}}$$

X,Y,Z = X,Y,Z ECEF coordinate system

4.4.2.2. Path Loss Range Determination

The RF path loss range shall be resolved for the following RF links:

1. Ground Control Station (GCS) to UAS Control Data Link
2. OCU to UCR Data Link
3. UCR to UGV Data Link

4.4.2.2.1. Link Configuration Data

The configuration details for each link shall be stored in user maintained records. The link configuration for a particular link shall be selected by the user during System Standby state. Each link configuration record shall contain the following data:

1. Link frequency (L_{freq}) MHz

2. Transmitter power output (P_{tx}) dBm
3. Receiver sensitivity (P_{rx}) dBm
4. Transmit antenna gain (G_{tx}) dBi
5. Receive antenna gain (G_{rx}) dBi
6. Transmission line losses between transmitter and transmit antenna (L_{tx}) dB
7. Transmission line losses between receiver and receive antenna (L_{rx}) dB
8. The height of the mobile antenna (h_m)
9. The height of the base antenna (h_b)

4.4.2.2.2. Path Loss Range Equations

For each link the Path Loss and ranges shall be calculated using the Hata path loss equations detailed in Hata Open, Hata Suburban, Hata Small City and Hata Large City. The configuration file shall state which equation to use.

Hata Open Model: This is for wide open areas with no obstructions of any sort including buildings, terrain, and trees.

Given:

L_{freq} = link frequency in MHz
f_{MHz}= center frequency in MHz
h_b = height of the base antenna (RMAX)
h_m = height of the mobile antenna (UGV)$a(h_m) = [1.1 \log_{10}(f_{MHz})\text{-}0.7]\ h_m - [1.56 \log_{10}(f_{MHz})\text{-}0.8]$; antenna height gain correction factor
P_{rx} = the receiver sensitivity in dBm
G_{tx} = Antenna gain of transmitter in dBm
G_{rx} = Antenna gain of receiver in dBm
G_{tot} = total gain of link antennas in dBm ($G_{tx} + G_{rx}$)
P_{tx} = Transmit power in dBm (1 W = 30 dBm)
$K = 4.78 [\log_{10}(f_{MHz})]^2 - 18.33 \log_{10}(f_{MHz}) + 40.94$, Environment correction factor, (ie suburban and open areas)

$d_{hata} = \text{antilog}_{10} \{[P_{tx} + G_{tot} - P_{rx} - 69.55 - 26.16 \log_{10}(L_{freq}) + 13.82 \log_{10}(h_b) + a(h_m) + K] / [44.9 - 6.55 \log_{10}(h_b)]\}$; the maximum radio transmission distance in meters

Hata Suburban Model: This is for Suburban areas where there are a few obstructions due to sparse buildings and slightly rugged terrain.

Given:
$K = 2 [\log_{10}(f_{MHz}/28)]^2 + 5.4$

$d_{hata} = \text{antilog}_{10} \{[P_{tx} + G_{tot} - P_{rx} - 69.55 - 26.16 \log_{10}(L_{freq}) + 13.82 \log_{10}(h_b) + a(\mathrm{h}_m) + K] / [44.9 - 6.55 \log_{10}(h_b)]\}$

Hata Small City Model: This is for city areas with high concentration of low lying buildings and uneven terrain.

Given:
$K = 0$

d_{hata} = antilog$_{10}$ $\{[P_{tx} + G_{tot} - P_{rx} - 69.55 - 26.16 \log_{10}(L_{freq}) + 13.82 \log_{10}(h_b) + a(h_m) + K] / [44.9 - 6.55 \log_{10}(h_b)]\}$

Hata Large City Model: This is for large urban areas with tall buildings or in mountainous terrain.

Given:
$a(h_m) = 3.2\,[\log_{10}(11.75\, h_m)]^2 - 4.97$
$K = 0$

d_{hata} = antilog$_{10}$ $\{[P_{tx} + G_{tot} - P_{rx} - 69.55 - 26.16 \log_{10}(L_{freq}) + 13.82 \log_{10}(h_b) + a(h_m) + K] / [44.9 - 6.55 \log_{10}(h_b)]\}$

4.4.2.2.3. GCS to UAS Control Data Link

1. Input = GCS_UAS_LNK_CFG (user selected)
2. Output = GCS_UAS_LNK_PLR (km)

Where,

GCS_UAS_LNK_CFG is the configuration of the Hata Model Parameters for computing d_{hata} between the GCS and the UAS.
GCS_UAS_LNK_PLR = d_{hata}

4.4.2.2.4. OCU to UCR Data Link

1. Input = OCU_UCR_LNK_CFG (user selected)
2. Output = OCU_UCR_LNK_PLR (km)

Where,

OCU_UCR_LNK_CFG is the configuration of the Hata Model parameters for computing d_{hata} between the OCU and the UCR.
OCU_UCR_LNK_PLR = d_{hata}

4.4.3. UCR to UGV Data Links.

4.4.3.1. Input

1. UCR_UGV*x*_LNK_CFG (user selected) where *x* is the UGV number

Where,

UCR_UGV*x*_LNK_CFG is the configuration of the Hata Model Parameters for computing d_{hata} between the UCR and selected UGV_x.

4.4.3.2. Output

1. UCR_UGV*x*_LNK_PLR (km) where *x* is the UGV number

Where,

UCR_UGV*x*_LNK_PLR = d_{hata}

4.4.3.3. UAS Maximum Operational Boundary

UAS Optimum Maximum Range

The maximum operational range boundary of the UAS that maintains the OCU-UCR link is defined as follows and made available to be displayed.

1. UAS_MAXR(km) = $\sqrt{a^2 - b^2}$

Where:

a = OCU_UCR_LNK_PLR
b = UAS Altitude

4.4.3.3.1. Altitude at UAS Optimal Maximum Range

The UAS altitude at the maximum operational range is defined as either:

1. Minimum Altitude = (Height above the horizon at maximum range) + (2 * worst case 1st Fresnel zone clearance) + (OCU altitude (MSL)).
2. Security Operational Altitude 610m (~2000ft) Above Ground Level (AGL).

Height above the Horizon: Height above the horizon at optimal maximum range =

$$\left(\frac{OCU\_UCR\_LNK\_PLR}{112.88}\right)^2$$

Worst Case 1st Fresnel Zone Clearance Determination: The 1st Fresnel zone clearance at any point P is given by:

$$F_1 = 17.3\sqrt{\frac{d_1 d_2}{f(d_1 + d_2)}}$$

Where:

F_1 = 1st Fresnel zone in meters
d_1 = distance to P from the OCU in meters
d_2 = distance of P from the UAS meters
For the worst case d_1 = d_2 = $\frac{OCU\_UCR\_LNK\_PLR}{2}$
f = frequency of RF signal in GHz

UGV Maximum Operational Range Boundary: The UGV maximum operational range boundary is defined as the UAS maximum range boundary plus 0.8*UGV Footprint Radius to ensure that the UAS operating area remains within communication limits of the OCU Comm-Repeater link.

4.4.3.3.2. UAS Optimal Operating Area

The UAS operating area is defined as the area of overlap of the UCR-UGV link ground footprints, and the UGV collaborative footprint, which lies within the UAS maximum operating boundary.

Operating Area Determination: Determination of the operating area is completed by the following process:

1. Determine UAS Altitude

2. Determine UGV footprint radius
3. Determine collaborative status of UGV’s
4. Determine collaborative communication footprint of UGV’s
5. Determine if the maximum UCR operating boundary imposes limits to the UGV collaborative communication footprint

UAS Operating Altitude: UAS Operating Altitude will be dependent on the mission type. Two altitudes are defined, one for maximum UGV operating area and a second for security of the UAS and will be user selectable.

Altitude for Maximum UGV Operating Area:
1. Determine the range to the furthest UGV from OCU.
2. Minimum UAS altitude Mean Sea Level (MSL) = (Height above the horizon at the range to the furthest UGV) + (2 * worst case 1st Fresnel zone clearance) + OCU altitude (MSL).

Altitude UAS Security: For security of the UAS, to minimize the risk of it being hit by small arms fire from the ground, the altitude will be set to 610m (~2000ft) AGL.

UGV Footprint Radius: Radius of ground footprint (Figure 16), UGVx$_{gfr}$ is defined as the $\sqrt{c^2 - d^2}$;
Where:
c = UCR_UGV_LNK range
d = UAS operating altitude (MSL) – UGV altitude

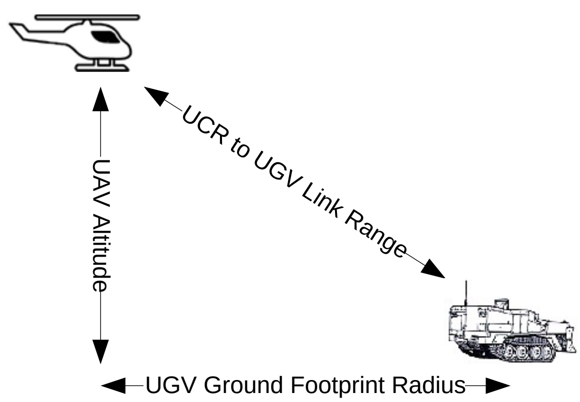


Figure 16. UGV Ground Footprint Radius

UGV Collaborative Status
Two UGV’s: Two UGV’s are defined as collaborative if their respective ground footprints overlap by 15% of the footprint radius e.g. distance between UGVx and UGVy < 1.85 * ground footprint radius (Figure 17).

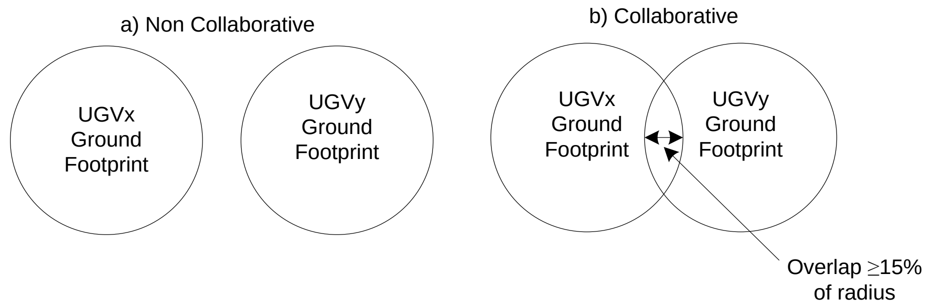


Figure 17. Two Vehicle Collaboration

Three or More UGV's: For three UGV's there are two cases of collaboration. Case (1) is defined as their respective ground footprints overlap such they form a circular triangle. Case (2) is defined as their respective ground footprints overlap such they form a convex lens shape, as per 2 UGV's (Figure 18).

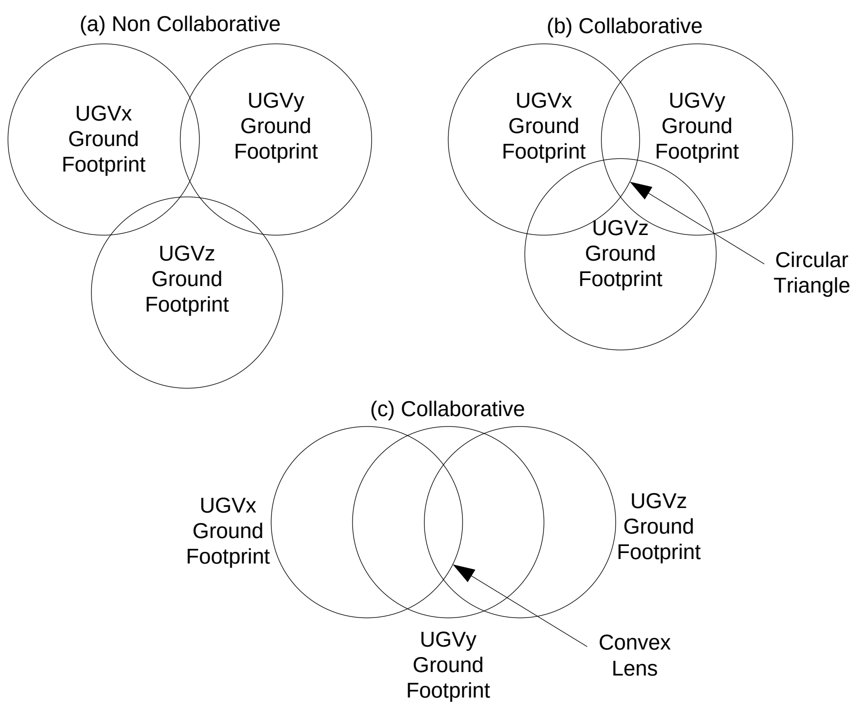


Figure 18. Three Vehicle Collaboration

The minimum overlap requirements for three vehicle collaboration are greater than 15% overlap between each collaborative vehicle as seen in Figure 18.

For more than 3 UGVs, only three will be selected, with one being the priority vehicle. This approach results in the largest the Collaborative Communication Footprint (CCF).

Determination of Collaboration: Determine the collaborative status of the UGV's.

1. Determine collaborative status of priority x UGV with-respect-to priority y UGV.

4.4.3.4. Collaboration Test for UGVx with-respect-to UGVy

1. Inputs
   a. UGVx position in ENU axis *(UGV1e, UGV1n)* = x_x, y_x
   b. UGVx ground footprint radius $(UGV1_{gfr}) = r_x$
   c. UGVy position in ENU axis *(UGV2e, UGV2n)* = x_y, y_y
   d. UGVy ground footprint radius $(UGV2_{gfr}) = r_y$
2. Process
   a. Determine distance, d_{12}, between the vehicles

$$d_{xy} = \sqrt{(x_y - x_x)^2 - (y_y - y_x)^2}$$

Collaborative test

If $d_{xy} \geq r_x + r_y$, then vehicles are non-collaborative, $COL_{xy}=False$

If $d_{xy} \leq \left\| r_x - r_y \right\|$, then the vehicles are 100% overlapped (i.e. one circle is contained within the other).

If $d_{xy} \leq r_x + r_y$, then calculate percent overlap.

$$\left(r_x + r_y \right) - d_{xy} = overlap$$

If overlap is $\geq r_y * 0.15$ then vehicles are collaborative, $COL_{xy}=True$

Selection of Vehicles from >3 Vehicles

Select the combination of three vehicles, from every three vehicle combination which includes the priority vehicle that meets the following collaborative condition and has the greatest sum of distances between their respective centers:

COLxy and *COLxz* and *COLyz* = *True* where x is the priority vehicle

Calculation of Intersection Points: The maximum number of ground vehicles in a collaborative status which will be supported is eight. The collaborative area is formed by the signal footprints of the ground vehicles which pass the collaboration test.

Find all the intersection points of all the collaborative circles with each other i.e. (1 & 2, 1 & 3, 2 & 3, etc.). There is the possibility of 54 intersections resulting from these calculations.
The circle number corresponds to the number of the UGV priority e.g. UGV 1 RF footprint radius = circle 1 radius.

4.4.3.5. Determination of Collaborative Points of Intersection

Calculate the distance to the radial from the position of UGVx (the radial is a line connecting the two points of intersection of the ground footprints).

$$d_{rxy} = \frac{\left(\mathrm{r_x}^2 - \mathrm{r_y}^2 + \mathrm{d_{xy}}^2 \right)}{2d_{xy}}$$

Calculate the angle between one point of intersection and the position of UGVx

$$\theta_{xy} = a\cos\left(\frac{d_{rxy}}{r_x}\right) rads$$

Calculate the angle between the position of UGVx and UGVy.

$$\sigma_{xy} = a\tan\left(\frac{y_y - y_x}{x_y - x_x}\right) rads$$

Note: Use *atan2* function to avoid divide by zero errors.

Calculate the intersection point coordinates $IP_{xy-1} = (x_{xy-1}, y_{xy-1})$ and $IP_{xy-2} = (x_{xy-2}, y_{xy-2})$

$$x_{xy-1} = x_x + \left(\cos\left(\sigma_{xy} + \theta_{xy}\right) r_x\right)$$
$$y_{xy-1} = y_x + \left(\sin\left(\sigma_{xy} + \theta_{xy}\right) r_x\right)$$
$$x_{xy-2} = x_x + \left(\cos\left(\sigma_{xy} - \theta_{xy}\right) r_x\right)$$
$$y_{xy-2} = y_x + \left(\sin\left(\sigma_{xy} - \theta_{xy}\right) r_x\right)$$

From the calculated intersections, and the known radii of the collaborative footprints the collaborative area can be determined.

Determine which of the calculated intersections are within the collaborative footprints.

$$d = \sqrt{(x_1 - x_{xy-1})^2 + (y_1 - y_{xy-1})^2}$$

If d < or = to r_1 then that intersection is within circle 1.
In order for an intersection to be collaborative, it must be within all of the circles (1- 8(max)).

4.4.3.6. Collaborative Points of Intersection Ordering

Given the points of intersection which lie within the collaborative area, they must then be put in a sequential order (clockwise, counterclockwise) so as to form a regular polygon.

The two circles which form the intersections that are within the collaborative area are also known.

The first intersection in the order is the intersection closest to the OCU location. This is determined by

$$d = \sqrt{(x_{\text{int}} - x_{ocu})^2 + (y_{\text{int}} - y_{ocu})^2}$$

The intersection with the smallest d is the closest to the OCU.

The next intersection in the sequence has to be one which has one of the same circles represented.

For example:
Intersection 1: formed by circle 1 and 2

Intersection 2: formed by circle 1 and 3 or 2 and 3 or any combination, as long as one of the circles is the same as the one in intersection 1.

The maximum number of collaborative intersections is eight.
This ordering puts the intersections into an order which can be used as waypoints, or used in determining the optimal position.

UGV Collaborative Communication Footprint: Determination of Sufficient Area
Area of minimum overlap for three or more vehicles

$$A_{\min} = \pi r^2$$

$$r = \frac{v^2}{9.81(\tan 20)}$$

Where:
v is the speed of the UAS in meters/second.
r is the radius of the turn

The actual area of overlap is as determined in Optimal Position Determination.
The actual area of overlap must be greater than A_{min}.

4.4.3.6.1. UAS Navigation

Two methods for UAS navigation will be determined; Waypoint Navigation and Optimal Position.

4.4.3.6.1.1. Waypoint Determination

One Circle: For a single UGV and corresponding circular RF footprint, 6 waypoints are defined. These waypoints are determined using ENU units. The coordinates will need to be converted back to latitude and longitude for output to the UAS operator.

The waypoints are defined in polar coordinates:
Waypoint 1: (r,0°)
Waypoint 2: (r,60°)
Waypoint 3: (r,120°)
Waypoint 4: (r,180°)
Waypoint 5: (r,240°)
Waypoint 6: (r,300°)

Where:
r is the radius of the footprint circle
The angle is in degrees.

These must be converted into ENU units for conversion to latitude and longitude coordinates:

$$x = r\cos\theta$$

$$y = r\sin\theta$$

Where:

θ is the angle
r is the radius of the footprint circle

Two Circles: For a Lens CCF 4 waypoints are defined as the two intersection points, DE, and the two midpoints of each side, D'E', (Figure 19). These waypoints are determined using ENU units. The coordinates will need to be converted back to latitude and longitude for output to the UAS operator.

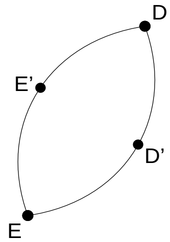


Figure 19. Waypoints for Convex Collaborative Communication Region

Priority 1

1. $IP_{12\text{-}1}COL$ AND $IP_{12\text{-}2}COL = True$
   a. $D = \left(x_{12-1}, y_{12-1}\right)$
   b. $D' = \left(\left(x_1 + r_1 \cos\sigma_{12}\right), \left(y_1 + r_1 \sin\sigma_{12}\right)\right)$
   c. $E = \left(x_{12-2}, y_{12-2}\right)$
   d. $E' = \left(\left(x_1 + \left(d_{12} - r_2\right)\cos\sigma_{12}\right), \left(y_1 + \left(d_{12} - r_2\right)\cos\sigma_{12}\right)\right)$

Priority 2

1. $IP_{13\text{-}1}COL$ AND $IP_{13\text{-}2}COL = True$
   a. $D = \left(x_{13-1}, y_{13-1}\right)$
   b. $D' = \left(\left(x_1 + r_1 \cos\sigma_{13}\right), \left(y_1 + r_1 \sin\sigma_{13}\right)\right)$
   c. $E = \left(x_{13-2}, y_{13-2}\right)$

$E' = \left(\left(x_1 + \left(d_{13} - r_3\right)\cos\sigma_{13}\right), \left(y_1 + \left(d_{13} - r_3\right)\cos\sigma_{13}\right)\right)$

Three Circles: For a Circular Triangle CCF 6 waypoints are defined as the three points, ABC, and the three midpoints of each side, A'B'C', (Figure 20). These waypoints are determined using ENU units. The coordinates will need to be converted back to latitude and longitude for output to the UAS operator.

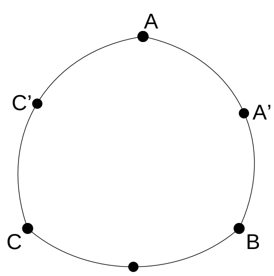


Figure 20. Waypoints for a Circular Triangle CCF

A, B and C Determination

1. Waypoint A = x_{12}, y_{12}
2. Waypoint B = x_{13}, y_{13}
3. Waypoint C = x_{23}, y_{23}

A', B' and C' Determination

1. A' Inputs
   UGV1 position in ENU axis (UGV1e, UGV1n) = x, y
   UGV1 ground footprint radius (UGV1gfr) = r
   Intersection point of UGV1-UGV2 footprints closest to UGV3, point A = (x_1, y_1)
   Intersection point of UGV1-UGV3 footprints closest to UGV2, point B = (x_2, y_2)

2. B' Inputs
   UGV3 position in ENU axis (UGV3e, UGV3n) = x, y
   UGV3 ground footprint radius (UGV2gfr) = r
   Intersection point of UGV1-UGV3 footprints closest to UGV2, point B = (x_1, y_1)
   Intersection point of UGV2-UGV3 footprints closest to UGV1, point C = (x_2, y_2)

3. C' Inputs
   UGV2 position in ENU axis (UGV2e, UGV2n) = x2, y2
   a. UGV2 ground footprint radius (UGV2gfr) = r2
   Intersection point of UGV2-UGV3 footprints closest to UGV1, point C = (x_1, y_1)
   Intersection point of UGV1-UGV2 footprints closest to UGV3, point A = (x_2, y_2)

4. Process
   a. $\theta_1 = a\tan 2\left(\frac{y_1 - y}{x_1 - x}\right)$
   b. $\theta_2 = a\tan 2\left(\frac{y_2 - y}{x_2 - x}\right)$

If

θ_1 And $\theta_2 > 0$
Or
θ_1 And $\theta_2 < 0$
Or
θ_1=0
Or

$\theta_2 = 0$

Then $\lambda = \theta_1 - \theta_2$

Else If

$\theta_1 > 0$
And
$\theta_2 < 0$
And
$|\theta_1| > (\pi/2)$ And $|\theta_2| > (\pi/2)$
Or
$\theta_1 - \theta_2 > \pi$

Then $\lambda = \theta_1 - (\theta_2 + (2*\pi))$

Else If

$\theta_1 < 0$
And
$\theta_2 > 0$
And
$|\theta_1| > (\pi/2)$
Or
$|\theta_2| > (\pi/2)$
And
$|\theta_1 - \theta_2| > \pi$

Then $\lambda = \theta_1 + ((2*\pi) - \theta_2)$

Else $\lambda = \theta_1 - \theta_2$

c. $\theta_3 = \theta_1 - \left(\frac{\lambda}{2}\right)$

d. Waypoint ' = x', y'

$$x' = x + (r\cos\theta_3)$$
$$y' = y + (r\sin\theta_3)$$

Four or More Circles: The waypoints for four or more circles will be the intersection points as calculated and ordered in Calculation of Intersection Points.

UAS Steering Update: Steering updates to the UAS operator shall be given at the rate of 0.5Hz to be completed.

4.4.3.6.1.2. Waypoint Navigation

These steps determine how the waypoint to be followed is selected.

Inputs: Six waypoints determined in section 3.6.1.3 or 3.6.1.2 or four waypoints from 3.6.1.4.

Known OCU location in ENU units
UAS position and heading, *UASe*, *UASn* and *UAShd*

Process

1. Convert the UAS magnetic heading to an angle in the ENU reference frame:
   a. $h = 90 - UAVhd$
   b. If *h*<0, then add 360, else leave as is.
   c. $h * \left(\frac{\pi}{180}\right) = hd$ in radians

hd in radians = UAS heading angle.

2. Determine the angle between the UAS and each waypoint zeta:
   a. If hd > $\left| a\tan 2\left(\frac{wpn - UAVn}{wpe - UAVe}\right) \right|$

      Then temp = $hd - a\tan 2\left(\frac{wpn - UAVn}{wpe - UAVe}\right)$

      Else temp = $a\tan 2\left(\frac{wpn - UAVn}{wpe - UAVe}\right) - hd$
   b. If temp > PI

      Then zeta = $2 * PI - |temp|$

      Else zeta = $|temp|$
3. Determine the distance from the UAS to each waypoints (*l*):
   a. $l = \sqrt{(wpe - UAVe)^2 + (wpn - UAVn)^2}$
4. Sort the waypoint by zeta, in ascending order.
5. Out of all the waypoints select the one with the smallest angle between the UAS heading and a waypoint.
6. Check for validity of waypoint (i.e. within the maximum UAS range). If valid, then continue to the waypoint location. Else, move to the waypoint with the next smallest angle and check for validity.
7. When the distance between the UAS and the waypoint < 50meters, change navigation to the next waypoint. (This value is subject to change based on the UAS characteristics)

4.4.3.6.1.3. Optimal Position

The optimal position is determined for use with rotary winged vehicles.

When the Optimal Operating Area is a circular triangle the Optimal Position is defined as:
$OptimalPosition = x_{op}, y_{op}$

Optimal Position Determination: Given: x and y coordinates for each of the intersections within the collaborative area

Where:

$$A = \frac{1}{2}\sum_{i=0}^{n-1} x_i y_{i+1} - x_{i+1} y_i$$

$$x_{op} = \frac{1}{6A}\sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i)$$

$$y_{op} = \frac{1}{6A}\sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i)$$

n = number of intersections that are within the collaborative area

Optimal Heading Determination: The optimal heading will be calculated based on the optimal position of the UAS and its relationship to the OCU location. The optimal heading has the tail of the UAS facing the OCU location. This eliminates any blockage of the signal by the engine and payload areas of the UAS.

Bearing from OCU to UAS in great circle terms:
θ = atan2 (cos (lat1) sin (lat2) − sin (lat1) cos (lat2).cos (Δlong), sin (Δlong) cos (lat2))

Where:
atan2 is of the form (x, y)
lat1 is the OCU
lat2 is UAS,
Result is +/- PI radians

This optimal heading is equal to the bearing from the OCU to the UAS.

Loss of Link Action: This describes the action taken by the UAS when link is lost with its ground control station.

Optimal Position Mode: The optimal position generated every thirty seconds will be stored in memory for 3-min. These waypoints in the memory will be recalled in order from the most recent to the least recent in the event of a loss of L-Band link. These waypoints will be followed until the link is restored. In the event of no recovery of the link, the vehicle will return to the “home” position.

Waypoint Navigation Mode: The same procedures used in Optimal Position Mode for reacquisition, waypoint memory storage, and for failure to reacquire will be used.

4.4.3.6.2. Link Status

The path loss determinations and the expected received signal strength will be compared with the actual received signal strength if a feedback loop is available.

The expected receive signal strength of the link is determined as follows:

$$\Pr x = -(\log_{10} d_{hata}(44.9 - 6.55\log_{10}(hb)) - Ptx - G_{tot} + 69.55 + 26.16\log_{10}(Lfreq) - 13.82\log_{10}(hb) - a(h_m) - K)$$

Where:
d_{hata} is the value calculated in Path Loss Range Equations for the model being used
$a(h_m)$ is the value calculated in Path Loss Range Determination for the model being used
K is the value calculated in Path Loss Range Equations for the model being used.

To verify the validity of the solution this value is compared with the average value over ten samples of the signal strength fed back to the OCU and is calculated as follows:

$$\%difference = \frac{calculated - actual}{actual} * 100\%$$

For a percent difference of <10%, the radius determined by the path loss models is valid. For a precent difference of >10%, the radius determined must be adjusted down by 10% of its current value.

If the calculated value results in a larger radius, no adjustments are necessary and the circles will be left as is.

L-Band Link: The radius of the L-band footprint will be calculated based on the Hatapath loss models. There are four models for the varying environments encountered.

WiFi Link Status: The radius of the S-band footprint will be calculated based on the Hata path loss models. There are four models for the varying environments encountered.

4.4.4. JCTE Technical Objectives

The LMS was originally developed to provide a generic capability supporting wireless network communication management for UMS. For the JCTE effort the LMS capability was tailored to support automating placement of the UCR Comm-Payload. This capability was targeted at establishing and maintaining effective communication via the UCR while reducing operator workload associated with managing the RMAX UAS. Technical objectives established for the LMS under the JCTE effort included the following:

- Determine Optimum Position for UAS with Communication Repeater Payload to establish and maintain BLOS communication.
- Determine Acceptable Region of Operation for UAS with Communication Repeater Payload to establish and maintain BLOS communication.
- Reduce Operator Workload associated with Operation of Communication Repeater

To meet the first objective the LMS had to be modified to take into account the UCR L-band uplink and S-band downlink. This was in addition to determining the effective communication regions for each of the UGVs. In order to determine the optimum location for placement of the RMAX UAS, the LMS had to compute the effective communication region for the OCU Comm-Package as well as for each of the UGVs operating in the UMS network. In one scenario the

UAS with Comm-Payload might be placed between the OCU and UGV area of operations. In this case, moving the UAS closer to the UGVs might improve the performance of the S-band communication link between the Comm-Payload and UGVs but degrade the L-band link between the OCU and UAS. The converse is true if the UAS is moved closer to the OCU and thus farther away from the UGVs. The LMS would automatically compute the optimum location for placement of the RMAX UAS with Comm-Payload based on position reports from all participating nodes in the network.

In addition to determining the optimum position for placement of the UAS with Comm-Payload the LMS would also compute an acceptable region of operation. This capability was originally developed to support fixed wing employment of the Comm-Payload. With a rotary wing platform like the RMAX the Comm-Payload can be positioned at a specific location (i.e. latitude/longitude/altitude) in space. With a fixed wing platform the air vehicle must be moving in order to produce lift and thus cannot support a position and hold. For this case, the LMS was designed to compute an acceptable region for effective communication. The boundary of that region was used to generate a dynamic flight path for the fixed wing air vehicle. A fixed number of waypoints were generated. Each waypoint was on the perimeter of the region computed as supporting acceptable communication for the UMS network. The waypoint nearest to the air vehicle was designated as the next fly-to waypoint for the air vehicle. Azimuth boundaries were utilized so that the air vehicle proceeded to the next identified waypoint that lay within its flight regime. In this manner the air vehicle would travel from waypoint to waypoint, thus proceeding around the boundary of the region for acceptable communications that supported the BLOS communications repeater capability.

4.4.5. JCTE Integration Effort.

The JCTE integration effort related to the LMS was focused in three main areas:

- Integrating the LMS Server to the JAUS backbone.
- Integrating the LMS GUI to the JAUS backbone.
- Integrating the RMAX GCS to the JAUS backbone.

As previously mentioned the LMS Server monitored position reports from all fixed and mobile participants in the UMS network and determined the optimum position for placement of the RMAX UAS with Comm-Payload. In order to perform this function the LMS was interconnected to the JAUS backbone (i.e. Ethernet network). The JCTE JAUS IDD [1] identified the details of all JAUS messages supported by the LMS. The LMS was on a wired Ethernet link to the JAUS backbone while other systems including the UGVs were operating over wireless interfaces. The LMS Server received a list of vehicles in the UMS network and monitored for updated position reports for the list of vehicles. The LMS Server would also report global information for use by the LMS GUI. These interfaces were developed and tested during the integration activities that led up to the JCTE Demonstration event.

Like the LMS Server, the LMS GUI was interconnected to the JAUS backbone via a wired Ethernet interface. The LMS GUI received global information from the LMS Server. This information was used to provide a COP with dynamically updated vehicle positions as well as displaying effective regions of communication and optimum position for the UAS with Comm-Payload.

In order to automate the process by which the RMAX UAS would fly to the LMS generated optimum position, the LMS generated optimum position had to be sent to the RMAX GCS. This was done via a custom interface between the RMAX GCS and JAUS backbone. A third party software module developed by Viking was procured and integrated with the RMAX GCS. This software module received a standardization agreement (STANAG) compliant waypoint and generated the necessary commands to send the UAS to the designated position. A separate software module was developed to translate a JAUS compliant waypoint into the STANAG compliant format. Optimum position waypoints generated by the LMS were sent to the RMAX GCS via the JAUS backbone. These messages were then converted into a STANAG message format for use by the GCS Controller and Viking software. The Viking software would then control the RMAX to the designated position.

4.4.6. Test and Evaluation.

Testing of the LMS was performed in a stage wise manner. The first stage of testing was performed in the lab with simulated inputs for the OCU, UAS, and UGV positions being generated and input to the LMS. During this phase the basic elements of the LMS algorithms and LMS GUI were tested and evaluated for correct performance. LMS configuration file operations (e.g. open, read, etc.) were also verified. User interfaces were refined early on to improve information displays based on operator feedback.

The next phase of testing was performed at the Robotics Lab located at Tyndall AFB using real UGVs with simulated UAS fly-outs being generated by the RMAX GCS. JAUS interfaces were tested and verified to be operating correctly. Map databases were loaded for the specified area of operations and correct display of vehicle positions and effective communication regions was verified. During this phase of integration testing it was verified that the RMAX GCS was receiving waypoints generated by the LMS Server. However, the RMAX was not flying at this time.

The third integration phase included additional hardware such as the AMRDEC robotic HMMWV and SPAWAR AUMS systems. Again, vehicle positions on LMS GUI displays were verified for correct location and presentation. RMAX UAS flight operations were performed at this time. During this phase of integration RMAX response to LMS generated waypoints was enabled. The RMAX correctly responded to and flew to the LMS generated waypoints while the UCR Comm-Payload supported BLOS communications between operators and UGVs operating in a networked environment. In order to safeguard against invalid waypoints, the JAUS to STANAG converter generated boundary conditions that effectively created a safe flight envelope in which the RMAX UAS operated.

4.4.7. Results.

The results of LMS integration and testing were favorable considering the LMS capabilities and limitations for a Phase 1 level system. The Phase 1 capabilities are further described in the results detailed in this section.

The LMS was effectively integrated to the JAUS backbone. A configuration file containing parameters of the communication equipment for each of the UMS network participants, to include OCU, GCS, UAS, and UGVs was read and processed by the LMS. Dynamic position

updates from each of the participants were read by the LMS and correctly displayed on the LMS GUI. Overlapping effective communication regions were computed by the LMS Server and displayed on the LMS GUI as expected. The RMAX GCS correctly received and responded to the optimum position reports from the LMS Server and when enabled the RMAX UAS flew to the LMS Server generated waypoints.

One deficiency that was noted was in the performance of the LMS generated regions of effective communication. It appeared to the operators that the created regions of effective communication for each vehicle were too large in diameter, thus leading one to believe that a UGVs wireless network interface transceiver would be able to communicate over a much larger distance that it could actually support. This result can be attributed to the three major factors described next.

First, the LMS Phase 1 capability computes path loss based on known basic path loss equations and equipment parameters (e.g. Tx power, Rx Sensitivity, antenna gains, etc.) for each participant. Several other path loss algorithms have been identified that would be more suitable to RF propagation characteristics for the 802.11 WiFi frequency and modulation characteristics. These algorithms were slated for inclusion during an LMS Phase 2 development effort.

Second, the LMS Phase 1 capability does not take into account terrain obstructions (e.g. shading by buildings, trees, foliage, etc.). This again was slated to be incorporated into the LMS as a Phase 2 capability. The LMS Phase 1 capability therefore considers all communication to be straight line-of-site with no obstructions. In reality the area of operations for the UGVs included trees and foliage that contributed to additional RF signal path loss.

Third, the LMS Phase 1 capability does not take into account antenna patterns and specifically the roll-off in elevation for a toroidal pattern as typically characteristic of a monopole antenna like those used on the UGVs. RF path loss is a function of gains and losses. Gains can essentially be associated with Tx power, receiver sensitivities, and antenna gains. For instance, if you increase your antenna gain, you should have a corresponding increase in effective communication range. RF losses can be attributed to cable/connector losses, general path loss, terrain shading (e.g. buildings, trees, foliage, etc.), atmospheric conditions, and in general from other RF noise sources. With respect to antenna patterns, for a monopole omni-directional antenna, as is typically used on a mobile vehicle such as a UGV, the antenna pattern is a toroid that is laid flat and thus uniform in the horizontal plane. In the elevation plane the antenna pattern, and associated gain, rolls off until you reach a null zone located top dead center over the zenith of the antenna. Some omni-antennas such as those used in blade antennas in airborne applications are specifically designed for a uniform elevation pattern to mitigate the effects of variations in air vehicle attitude (i.e. roll, pitch, and yaw). Again, most monopole antennas used in UGV applications are not of the airborne (i.e. uniform elevation pattern) type. The geometry of the JCTE effort was such that the RMAX UAS flew overhead over the area of UGV operations. This resulted in significantly high look-up angles in elevation between a UGV and the RMAX carrying the Comm-Payload. These high look-up angles corresponded with significantly losses in antenna gain in the H-plane. This could be mitigated in a number of ways as detailed in the recommendations. For LMS improvements, the antenna patterns could be incorporated into the path loss equations to achieve a more realistic response from the system.

4.4.8. Recommendations.

Based on lessons learned during JCTE experimentation and demonstration the following recommendations have been made to improve the performance of the Phase 1 LMS capabilities:

- Incorporate additional path loss algorithms suited for cellular and WiFi communication waveforms and frequencies.
- Add real time operator interface for modification of system parameters (e.g. adjust fade margin).
- Add dynamic models that account for antenna pattern effects resulting from changes in vehicle attitude.
- Incorporate digital terrain elevation data (DTED) to account for terrain shading.
- Perform additional test/evaluation on automatically generated flight path associated with an acceptable region of communication.
- Test effectiveness of LMS with a Communication Repeater carried by a fixed wing platform.

As previously mentioned, the Phase 1 LMS capability incorporated a basic RF path loss equation. Other path loss equations have been identified that provide more accurate results for cellular technologies. It is believed that these equations would be more representative of the communication characteristics associated with 802.11 WiFi and thus would provide more accurate results in determining effective communication regions for a UMS wireless network. These equations could easily be incorporated into the LMS Server. An interface could be provided on the LMS GUI to allow for operator selection of the appropriate path loss equation based on the type of communication technology being used.

It is also recommended that a real-time interface be added to the LMS GUI to provide for operator input. This interface would allow for several things to include selection of an appropriate path loss equation as well as to adjust the system fade margin. Commands made by an operator at the LMS GUI would be sent to the LMS Server over the JAUS backbone to affect the desired response. The operator would also be able to select between optimum position mode or acceptable region flight path mode dependent on the UAS type being used, fixed wing or rotary wing. Other operator inputs might allow for real-time adjustment of configuration parameters like antenna gains or transmit powers.

Dynamic antenna models should be included in the LMS. These models would account for variations in antenna patterns based on vehicle attitude and antenna orientation. Corresponding gains would then be computed by the LMS to account for roll-off in signal strength as a function of elevation look-up angles between a UGV and a UAS carrying a Comm-Payload such as demonstrated in the JCTE effort.

DTED data should also be incorporated into the LMS. The LMS would be modified to utilize DTED information to determine RF propagation shading due to terrain (e.g. hills, ditches, etc.). The LMS Phase 1 capability does not make use of DTED information and thus works in an ideal environment where there are no hills, river valleys etc. To be useful, the LMS needs to account for terrain obscurations to RF propagation.

Additional testing should be performed on the LMS acceptable region profile using a fixed wing or rotary wing UAS to validate LMS performance in this mode. The acceptable region for communication is used as the basis for generating the waypoints that create a flight path for the UAS. Following the LMS generated flight path a UAS will remain within the computed region for acceptable communication supporting UMS networked operations.

4.4.9. Pointing Algorithms

The pointing algorithms originated from the computer-aided fire control (CAFC) which began in January 2005 as a technology program to develop technologies with potentially immediate application to remote weapons operation. The focus of CAFC was to develop technologies to reduce warfighter workload and to improve remote weapon performance by automating the targeting of a weapon. CAFC included a software interface that utilized a ballistics library which provided ballistic corrections given the physical properties of the bullet and atmospheric conditions. It also employed a pointing algorithm which computes the azimuth to target of a turret given the GPS coordinates of the turret and the target. JCTE is utilizing the pointing algorithm aspect of CAFC to target potential threats using AFRL's Defender platform, then adding upon CAFC capability by passing targets between unmanned systems.

The pointing algorithm allows a collaborative team to effectively monitor a threat or area of interest by slewing a turret, which may have a camera or a weapon attached, to a specified target. This algorithm computes the azimuth to target by taking into account the yaw, pitch, and roll of the vehicle on which the turret is mounted and the GPS coordinates of the turret and the target. The pointing algorithm is an effective method to use for the targeting and overwatch of threats, but because it assumes that a bullet will fly in a straight line, it is not accurate enough to provide precise engagement capabilities.

Precise engagement would be available in a collaborative environment by using the ballistic library algorithms. The ballistic library algorithms compute corrections based on the physical characteristics of the weapon and the round, and on the present atmospheric conditions. Physical properties such as the mass, diameter, form factor, and muzzle velocity are combined with atmospheric conditions such as temperature, pressure, humidity, altitude, and crosswind to produce a superelevation of the bore in order for the bullet to hit the target.

The ballistic library has been validated with a variety of projectiles and velocity regimes. It has been validated at supersonic velocities with a 7.62mm round at distances from 100m to 800m, at nearly sonic velocities with a MK-19 40mm grenade launcher from 100m to 300m, and at subsonic velocities with a FN303 less-than-lethal projectile from 10m to 100m. In a collaborative environment this would allow precision engagement from a variety of platforms.

In the demonstration at Tyndall Air Force Base, overwatch and targeting of a potential threat was demonstrated using two Defender UGVs. In a truly collaborative environment the ability to target and engage threats could be realistically utilized by UGVs, UASs, and unmanned turret emplacements. In order to achieve precise engagement, systems would need to be equipped with sensors capable of accurately detecting the distance to a target (Figure 21). This could be achieved by utilizing a laser-based distance sensor or DGPS. The effectiveness of the pointing algorithm is also dependant on the accuracy of the GPS system being utilized.

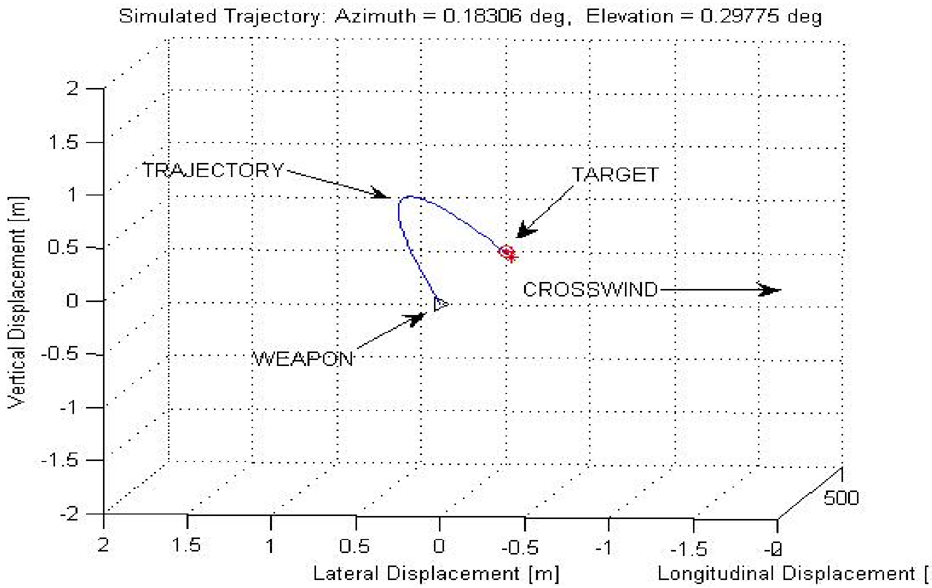


Figure 21. Simulated Trajectory

4.4.9.1. Algorithm Description

The pointing algorithm uses the GPS coordinates and orientation of the vehicle along with the GPS coordinates of the target to compute the pan and tilt commands. This is done by first computing the vector between the weapon and the target in local Cartesian coordinates using the following formula.

$$X_G = R_E(\alpha_{TARGET} - \alpha_{WEAPON})$$
$$Y_G = R_E(\beta_{TARGET} - \beta_{WEAPON})\cos\alpha_{WEAPON}$$
$$Z_G = Z_{TARGET} - Z_{WEAPON}$$

Here R_E represents the radius of the Earth measured in meters, α represents the latitude in radians, and β represents the longitude in radians. The subscripts indicate whether the variable in question represents the weapon or the target. Therefore, these coordinates (X_G, Y_G, Z_G) represent a line-of-sight targeting vector V_G in the global frame (North-East-Down) measured in meters. This vector can also be expressed in the local vehicle frame (Forward-Right-Down) in term of the components $V_L = \begin{bmatrix} X_L & Y_L & Z_L \end{bmatrix}^T$. These vectors are related by the rotation matrix R_{GL} as shown.

$$V_G = R_{GL} V_L$$

Here the matrix R_{GL} is a function of the roll ϕ, pitch θ, and heading ψ of the vehicle and can be written explicitly in terms of the sines and cosines of these angles as shown.

DISTRIBUTION STATEMENT A: Approved for public release; distribution unlimited.
88ABW-2013-2501, 28 May 2013

$$R_{GL} = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix}$$

Since R_{GL} is an orthogonal matrix, we can easily compute its inverse R_{LG} by simply forming the transpose the original matrix. Therefore, the following relationship between R_{GL} and R_{LG} exists.

$$R_{LG} = R_{GL}^{-1} = R_{GL}^{T}$$

The target vector in the local frame can now be computed using the following expression.

$$V_L = R_{LG} V_G$$

In order to simplify calculations later, the target V_L is normalized to produce the unit vector U_L as shown.

$$U_L = \frac{V_L}{|V_L|} = \begin{bmatrix} x_L \\ y_L \\ z_L \end{bmatrix}$$

A unit vector, U_W, is defined to be aligned with the axis of the weapon. For simplicity, the weapon coordinate system is defined so that its x-axis coincides with the vector U_W. Therefore, U_W can be written as shown.

$$U_W = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

The local vehicle frame and the weapon frame are related by the pan and tilt angles between the vehicle and the turret. Using these pan and tilt angles, the matrix R_{LW} can be formed that transforms vectors in the weapon frame to vectors in the local frame. The matrix R_{LW} can be expressed in terms of the sines and cosines of the pan and tilt angles as shown.

$$R_{LW} = \begin{bmatrix} C_P C_T & -S_P & C_P S_T \\ S_P C_T & C_P & S_P S_T \\ -S_T & 0 & C_T \end{bmatrix}$$

The goal is to find pan and tilt angles such that matrix R_{LW} will align the unit vector U_W with the unit vector U_L such that they are related by the following expression.

$$U_L = R_{LW} U_W$$

Substituting the explicit expressions for U_L, R_{LW}, and U_W, the following expression is obtained for the unit vector U_L in terms of the pan and tilt angles.

$$U_L = \begin{bmatrix} x_L \\ y_L \\ z_L \end{bmatrix} = \begin{bmatrix} C_P C_T \\ S_P C_T \\ -S_T \end{bmatrix}$$

The pan and tilt angles can now be computed using the following expressions

$$T = -\sin^{-1}(z_L)$$
$$P = \tan^{-1}(y_L / x_L)$$

By commanding the weapons turret to these values for pan and tilt, the weapon will be aligned with the line-of-sight vector connecting the vehicle platform and the target location.

4.4.9.2. Interim Findings/Lessons Learned

The JCTE efforts demonstrated that the implementation of the pointing algorithms to track, target, and pass targets to another system can be accomplished without any serious modifications to existing systems. This provides a significant capability increase for monitoring potential threats and reducing the kill chain process. It has also been shown that the ability to precisely engage a threat using the ballistics library will require sensor upgrades and is increasingly complex due to sensitivities of the algorithms to small variations in turret position and weapon calibration. Small variations caused by improper turret installation or slightly faulty sensor readings from the vehicle orientation cause increasing variability for ballistic library algorithms on moving vehicles. A more feasible approach in the near future is to implement the ballistic library in a force protection scenario using stationary turrets.

5. JCTE INTEGRATION

The initial task of integrating all the individual systems involved establishing a common communication interface, which included all the messages (commands and data), and the communication scheme to be used. The messaging scheme selected was that specified under the JAUS Reference Architecture version 3.2 [2]. A JCTE JAUS IDD was draftedand included (1) the communication scheme (transport protocol, network configuration and wireless communications setup), and (2) the details of all the JAUS messages supported by each system (see Section 5.1). The goal was for each system to be compliant with the IDD, tested separately using SPAWAR’s MOCU, and then tested together with all the other systems.

The second task of integration was for each system to be compliant to the JAUS IDD. This required the ground systems to be tested separately using the MOCU. This reduced the time needed during integration for message debugging. However, it took all three integration sessions to fully test most of the JAUS messages since messages were implemented based on their relative importance to JCTE (see Section 5.2).

The third task of integration was frequency management. This entailed determining all the radio frequencies used by all the systems and checking for interference. The RF links used included (1) 2.4GHz wireless Ethernet link, (2) 1.8GHz L-band Communications Repeater long range link, (3) RMAX R/C link, (4) RMAX WeControl link, (5) AUMS helicopter R/C link, and (5) AUMS control link. This task was addressed in the IDD and tested during the integrations sessions.

The fourth task was bandwidth utilization management for each radio link. All but the first two RF links above are point to point. The 2.4 GHz wireless Ethernet link and the 1.8GHz L-band link pass networked data. It thus becomes important to measure the bandwidth required by each ground system and verify that the links can support the bandwidths. In this case, the critical factor was the video passed back by the ground vehicles. This final task was addressed during the integrations sessions.

The four tasks explained above are necessary steps to prepare for the integration sessions. The integration sessions were scheduled based on the readiness of all the systems. There were three integration sessions with the last one culminating with the technology demonstration.

5.1. Interface Design Document

The IDD defines the JAUS interface to each of the functional components of the JCTE system. This includes all the JAUS messages handled by each interface. Additionally, the protocol for using specific messages is discussed and illustrated.

5.1.1. Wireless Communications and Network Configuration

The communications protocol for the link from the OCU to the ground vehicles and air vehicles is 802.11b/g wireless Ethernet. The radio hardware used is the Esteem 195Eg. There is a network of Esteem APs with a Root AP (AP1) and several Static Repeater APs (AP2). AP1 is tied to the C2 which includes the OCU(s), the GPS Base Station, and the Comm Repeater Base PCU (CR-

B). The AP2 radios are distributed to provide the best coverage of the operational areas around the airbase.

In order to extend the operation of the UGV and UAS, the Comm Repeater uses a 1.8 GHz L-band link connecting the CR-B and the Comm Repeater Remote PCU (CR-R). The RMAX uses an Esteem AP Bridge and acts as a Dynamic Repeater AP (AP3) to relay information to and from the UGVs and UASs, as depicted in Figure 22. Figure 23 depicts the communication scheme with IP addresses.

The Esteem APs are configured as an AP Bridge with the Comm Repeater mode ON. For the AP1, the repeater peer list includes the Media Access Control IDs (MAC IDs) of all AP2s. Each (AP2) includes the MAC ID of the AP that it is directly connected to (whether it is AP1 or another AP2). For the AP3 on RMAX, the repeater peer list is empty.

The UGVs and the AUMS UAS use Esteem radios configured as an Ether-station. This gives it a client radio behavior capable of roaming seamlessly through the different APs. Since it is a client, it has to be associated directly to a wired Ethernet port on a device on the vehicle side. Table 10 describes the configuration settings for the radios and Table 11provides the IP addresses for the radios.

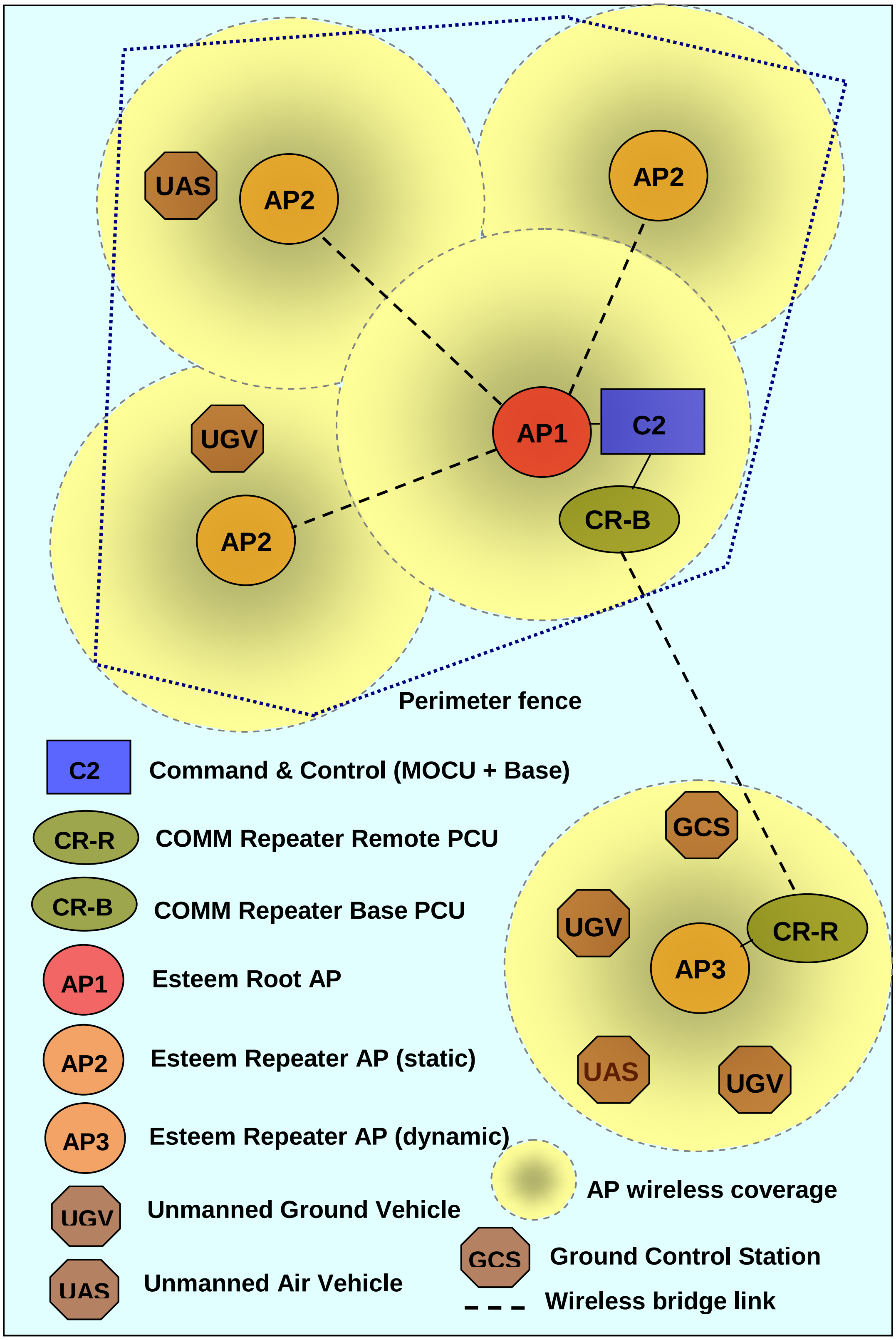


Figure 22. Wireless Communications Coverage

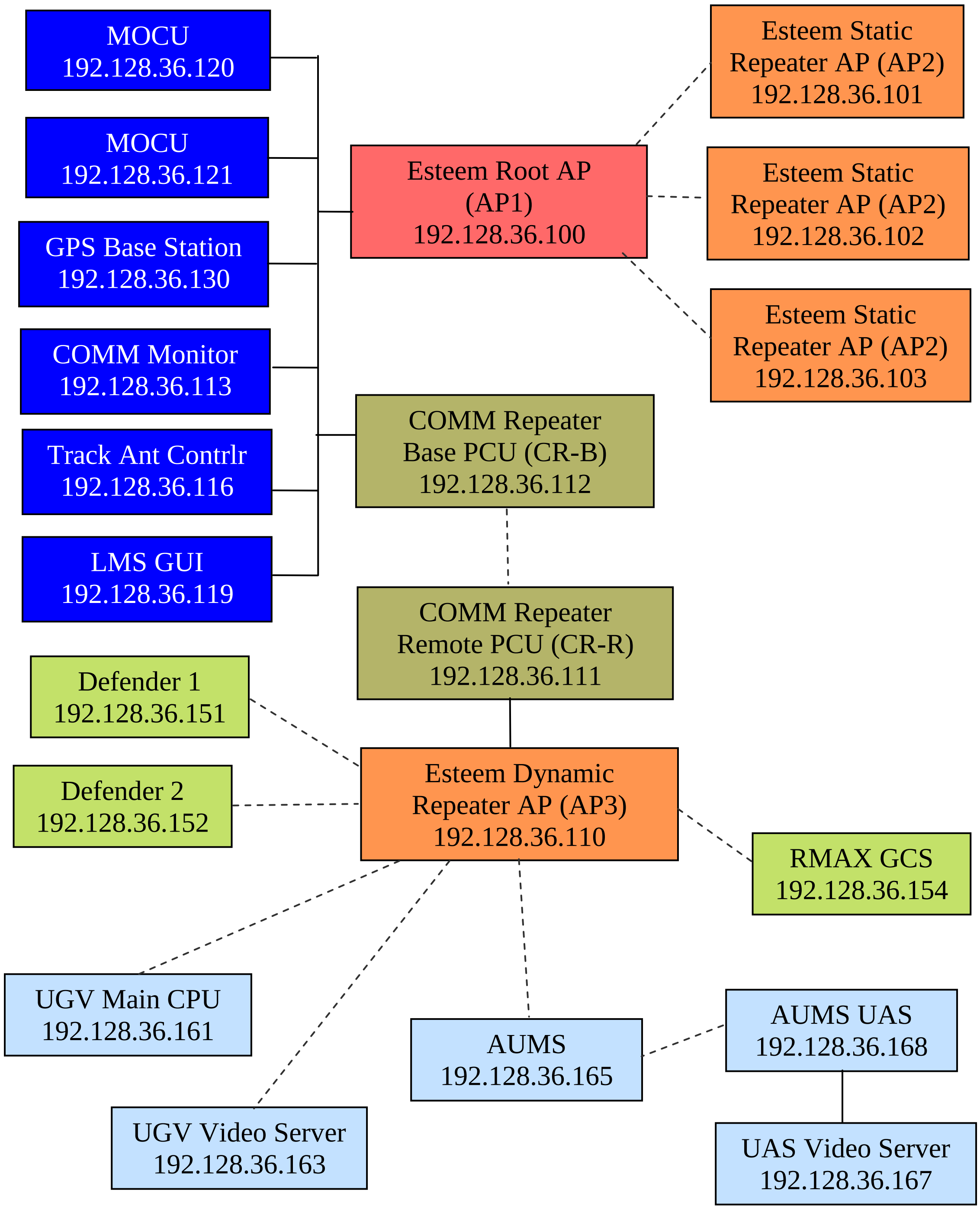


Figure 23. Network IP Address Scheme

Table 10. Configuration Settings

	Radio Mode	Ip Address	SSID*	Channel**	Security	Repeater Mode	Root Mode	Repeater Peer List
Root AP (AP1)	AP Bridge	192.128.36.100	roccocom	11	OFF	ON	YES	MAC ID of all associated repeater AP***
Static Repeater AP (AP2)	AP Bridge	192.128.36.101 – 105	roccocom	11	OFF	ON	NO	MAC ID of connected AP (AP1 or AP2)
Dynamic Repeater AP (AP3)	AP Bridge	192.128.36.110	roccocom	11	OFF	ON		
Etherstation (Client)	Etherstation	None	roccocom					

* Same SSID used in JE (can be changed)

** If problems arise, use channel 1 or 6

*** Do not include dynamic AP (AP3)

Table 11. IP Address Assignments

Name	IP Address	Remarks
Root AP (AP1)	192.128.36.100	
Repeater AP #1 (AP2)	192.128.36.101	
Repeater AP #2 (AP2)	192.128.36.102	
Repeater AP #3 (AP2)	192.128.36.103	
Reserved	192.128.36.104 - 109	Reserved for additional static Esteem APs
Dynamic Repeater AP #3 (AP2)	192.128.36.110	
COMM Repeater Remote PCU (RC-R)	192.128.36.111	
COMM Repeater Base PCU (RC-B)	192.128.36.112	
COMM Monitor PC	192.128.36.113	
COMM Monitor PC VM	192.128.36.114	
Tracking Antenna Controller	192.128.36.116	
Reserved	192.128.36.115, 117, 118	Reserved for devices to be connected directly to Base PCU (RC-B) such as computer that monitors status of COMM Repeater
LMS GUI PC	192.128.36.119	
MOCU (#1)	192.128.36.120	
MOCU (#2)	192.128.36.121	
Reserved	192.128.36.122 – 129	Reserved for additional MOCU, other OCU, or computers connected on Command & Control C2 network side
GPS Base Station	192.128.36.130	
Defender 1	192.128.36.151	
Defender 2	192.128.36.152	
Defender 3	192.128.36.153	Back-up DEFENDER
RMAX RJIM / GCS	192.128.36.154	laptop running RJIM, LMS, and RMAX NAV software
Reserved	192.128.36.155 – 159	Reserved for other AFRL UGVs
AUMS Radio	192.128.36.160	Reserved
AUMS Host UGV	192.128.36.161	Main CPU
AUMS Host UGV Axis Video Server	192.128.36.163	
Reserved	192.128.36.164	Reserved for other SPAWAR UGV/UAS
AUMS	192.128.36.165	
AUMS UAS	192.128.36.168	
AUMS UAS Video Server	192.128.36.167	
Reserved	192.128.36.36.169	Reserved for other SPAWAR UGV/UAS

5.1.2.

Some of the individual systems are already JAUS RA v3.2 compliant. This includes the Defender platforms and the MOCU. The specific details of all these messages as well as the JCTE specific user defined messages are described in the following sections.

5.1.2.1. RMAX UAS / COMM Repeater

There are two main JAUS components for the RMAX COMM Repeater system. The first is the lower level component which directly interfaces to the RMAX GCS software. This component is called RMAX. The OCU can control the RMAX UAS directly by sending commands to the RMAX Global Waypoint Driver (GWD) or Global Vector Driver.

The second component is called the Link Management System. The LMS will be used to calculate the optimal location of the COMM Repeater (located on the RMAX UAS) to be able to link all the ground vehicles. The LMS requires periodic position, orientation and velocity information from the ground vehicles namely, DEFENDER 1, DEFENDER 2, and AUMS Host UGV. The LMS also requires position feedback from the COMM repeater. This means that the LMS will interface to the JAUS Waypoint Driver of the RMAX. Figure 24 illustrates the JCTE JAUS configuration of the RMAX UAS and LMS, AUMS Host UGV, DEFENDER 1 and 2 UGV, and the MOCU.

RMAX Component: The RMAX component will be the primary JAUS interface to the RMAX UAS. This component is responsible for handling status data from the RMAX (e.g. RPM, battery voltage, position, orientation, velocity) and executing commands (e.g. waypoint and heading, or speed and heading).

The RMAX component supports messages for the following services:

1. Node Manager
2. Global Pose Sensor
3. Velocity State Sensor
4. Global Waypoint Driver
5. Visual Sensor (dependent on availability of RMAX video)
6. Communicator (dependent on link data availability)

1. Node Manager [2, 3] for dynamic discovery messages.

2. Global Pose Sensor [2]
   - Query Global Pose
   - Report Global Pose – all position and orientation data from the RMAX are available and can be requested through a service connection or event.

3. Velocity State Sensor [2]
   - Query Velocity State
   - Report Velocity State – all velocity data from the RMAX are available and can be requested through a service connection or event.

4. Global Waypoint Driver [2]
   - Set Global Waypoint – the OCU can command the RMAX to go to a waypoint
   - Set Travel Speed – the OCU can command the RMAX to go to a waypoint with a desired speed

- Query Global Waypoint – the OCU can request for the current waypoint the RMAX is negotiating. This is helpful as feedback when the LMS commands the RMAX directly to execute a waypoint.
- Query Travel Speed – the OCU can request for the current travel speed used by the RMAX in negotiating a waypoint. This is helpful as feedback when the LMS commands the RMAX directly to execute a waypoint.
- Query Waypoint Status
- Report Global Waypoint
- Report Travel Speed
- Report Waypoint Status – refer to Defender IDD v2.0
- Set RMAX Command – User Defined with Command Code E100h
    - This is used to execute higher level RMAX navigation functions (Table 12).

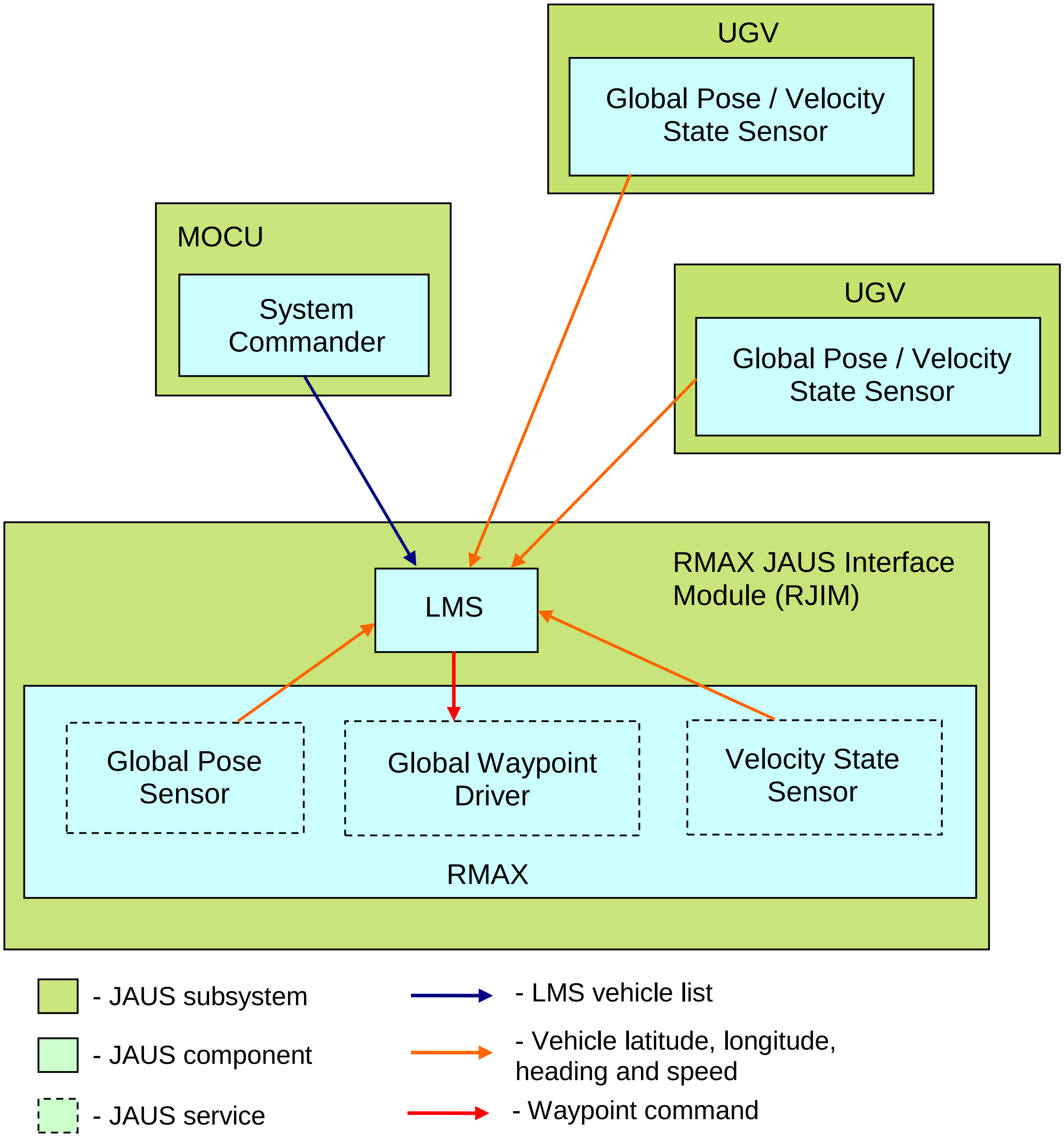


Figure 24. JCTE JAUS Configuration

Table 12. JAUS Byte Field Population for Message E100h: RMAX Commands

Field #	Name	Type	Units	Interpretation
1	Command ID	Byte	N/A	0: no action 1: land 2: launch 3: hover / loiter 4: continue to next waypoint 5 – 255: reserved

LMS Component: The LMS component acts as a higher-level planner that sends waypoints directly to the GWD. The required inputs to the LMS include position, velocity, and orientation data from the RMAX and from all the UGVs to be covered by Comm repeater wireless link. There are user-defined messages to allow the OCU to specify the list of UGVs to be included called the LMS Vehicle List. From a JAUS command hierarchy, the LMS controls the GWD. In order to use the LMS, the OCU takes control of the LMS and then the LMS takes control of the GWD. The OCU can also take control of the GWD directly by releasing the LMS which in turn releases the GWD.

The LMS supports messages for the following services:

1. Node Manager
2. Global Pose Sensor
3. Velocity State Sensor
4. Link Management Service (user-defined)

1. Node Manager – Refer to JAUS R.A. 3.2 and OPC 2.75 for dynamic discovery messages.

2. Global Pose Sensor [2]

- Query Global Pose
- Report Global Pose – LMS will set-up a service connection to each vehicle requesting for this message at 1 Hz. The data needed includes:
  - current vehicle latitude
  - current vehicle longitude
  - current vehicle altitude
  - current vehicle yaw (heading)

3. Velocity State Sensor [2]

- Query Velocity State
- Report Velocity State – LMS will set-up a service connection to each vehicle requesting for this message at 1 Hz. The data needed includes:
  - current velocity along the vehicle X-axis (along heading)

4. Link Management System (LMS) – User-Defined component

- Code F520h: Set LMS Vehicle List (Refer to Table 13)
  This is used to set the list of unmanned vehicles to be covered by the COMM repeater. The list includes the subsystem IDs. The LMS already has a vehicle list loaded through a configuration file. Setting a new vehicle list overrides the existing list as long as the subsystem that issues the *Set LMS Vehicle List* command has control of the LMS. This message requires an Acknowledged/Not Acknowledged (ACK/NAK) request from the OCU. If the requested vehicle list includes an invalid subsystem ID, a NAK is sent back to represent an error. This command can be sent with the LMS either in Standby or Ready.

Table 13. JAUS Byte Field Population for Message F520h: Set LMS Vehicle List

Field #	Name	Type	Units	Interpretation
1	Number of Vehicles, n	Byte	N/A	0: not used 1 – 255: valid
2	Vehicle 1 Subsystem ID	Byte	N/A	
3	Vehicle 2 Subsystem ID	Byte	N/A	
n + 1	Vehicle n Subsystem ID	Byte	N/A	

- **Code F521h: Query LMS Vehicle List**
  This is sent by the OCU to request for the current vehicle list that is set on the LMS.

- **Code F522h: Report LMS Vehicle List**

This is sent in response to the Query LMS Vehicle List. The vehicle data includes the vehicle subsystem ID and the availability of status data (position and velocity data) (Table 14).

Table 14. JAUS Byte Field Population for Message F522h: Query & Report LMS Vehicle List

Field #	Name	Type	Units	Interpretation
1	Number of Vehicles, n	Byte	N/A	0: not used 1 – 255: valid
2	Vehicle 1 Subsystem ID	Byte	N/A	0: not used 1 – 255: valid
3	Vehicle 1 Data Status	Byte	N/A	0 – No status, 1 – status available
4	Vehicle 2 Subsystem ID	Byte	N/A	0: not used 1 – 255: valid
5	Vehicle 2 Data Status	Byte	N/A	0 – No status, 1 – status available
2n	Vehicle n Subsystem ID	Byte	N/A	0: not used 1 – 255: valid
2n + 1	Vehicle n Data Status	Byte	N/A	0 – No status, 1 – status available

- **Code F523h: Add LMS Vehicle**
  This adds a single subsystem to the existing LMS Vehicle List. This message requires an ACK/NAK request from the OCU. If the requested vehicle has an invalid subsystem ID, a NAK is sent back to represent an error. This command can be sent with the LMS either in Standby or Ready.

- **Code F524h: Delete LMS Vehicle**
  This deletes a single subsystem from the existing LMS Vehicle List. This message requires an ACK/NAK request from the OCU. If the requested vehicle has an invalid subsystem ID, a NAK is sent back to represent an error. This command can be sent with the LMS either in Standby or Ready.

- **Code F525h: Clear LMS Vehicle List**
  This deletes all the subsystems from the existing LMS Vehicle List.

- **Resume** - This activates the LMS to start outputting waypoints to the GWD.

- **Standby** - This stops the LMS from outputting waypoints to the GWD.

5.1.2.2. Defender UGV

There are two parts to the JAUS interface for Defender 1 and 2 (Figure 25). The first involves the messages and protocol that have already been implemented, tested and used during past joint experiments. This is documented in a separate Defender IDD (version 2.0) [4].

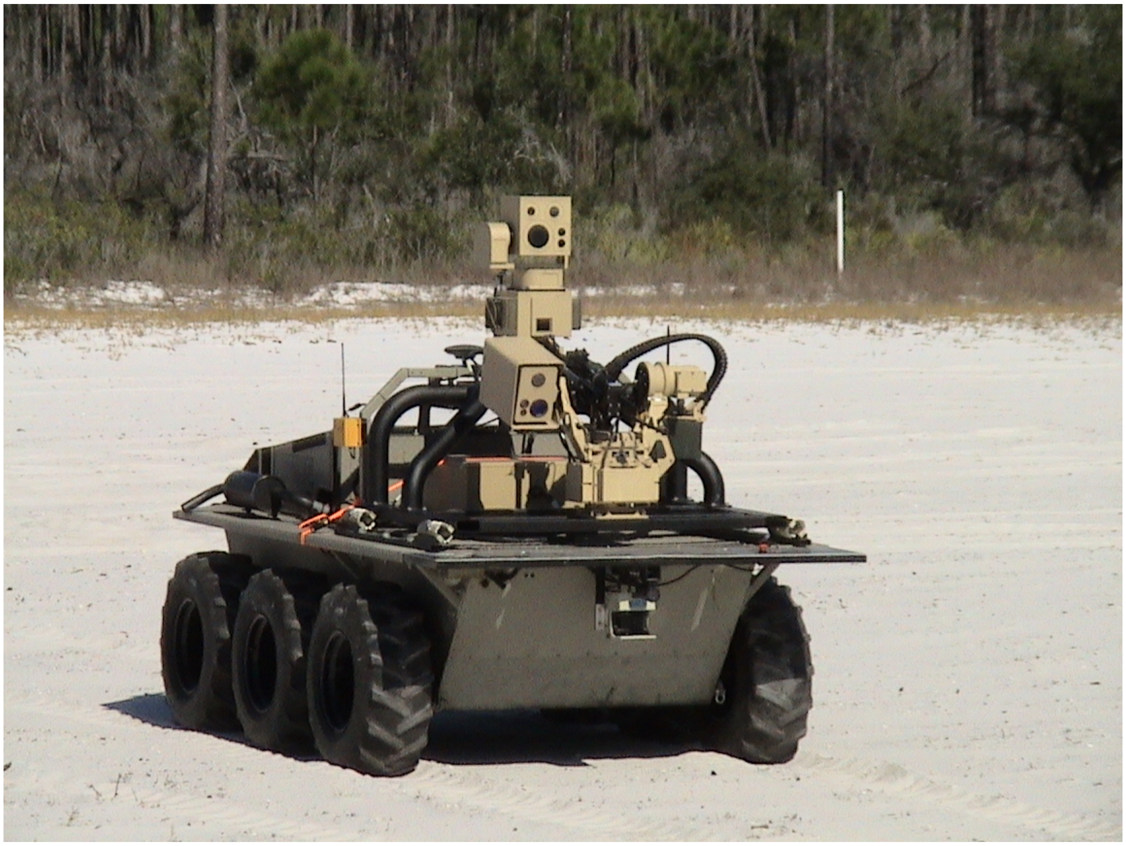


Figure 25. Defender UGV 1 &2

The second part includes all the supplemental services and messages added for the JCTE. This includes messages for Teaming and Target Detection System services. The following services outlined in (Table 15) are supported by the Defender:

Table 15. Fire Defender JAUS Mapping

1	Node Manager	Refer to Defender IDD v2.0, JAUS RA v3.2 and OPC 2.75
2	Primitive Driver	
3	Velocity State Driver	
4	Reflexive Driver	
5	Global Waypoint Driver	
6	Global Pose Sensor	
7	Velocity State Sensor	

8	Visual Sensor	
9	Audio Sensor	
10	Range Sensor	
11	Light Sensor	
12	Base Position Sensor	
13	Weapon Fire Control	Refer to Defender IDD v2.0 and JAUS RA v3.2
14	Weapon Watch	
15	Intruder Detection System	Refer to Defender IDD v2.0 and JAUS RA v3.2 This will be replaced by Target Detection System
16	Target Detection System	User Defined (see explanation below) This is used for detection reporting and vehicle track passing and pointing
17	Teaming	User Defined Implementation will be limited to formation of teams and team actions will be defined later

5.1.2.2.1. Target Detection and Track Passing/Pointing.

For the JCTE, the Defender vehicles work cooperatively in assessing and engaging potential targets. Both Defenders have a Target Detection System (TDS). Each UGV is capable of setting targets either through automatic detection (e.g. using the Weapon Watch) or by the operator using the OCU. The targets can be passed on to the other UGV by the operator sending a *Look At* command to the other UGV. The *Look At* command is a user-defined message under the Visual Sensor. The Reconnaissance, Surveillance, and Target Acquisition (RSTA) and the Remote Operated Weapons Systems (ROWS) components on the Defender support the Visual Sensor service and the TDS service.

(Figure 26 through Figure 29) illustrate the message flow and activities that happen during certain detection scenarios. It should be noted that the target detection and track passing is initiated by the operator. This is to satisfy current operational requirements (1) control of the visual sensor is needed to move the camera, (2) weapon fire control safety protocol requires an operator in the loop.

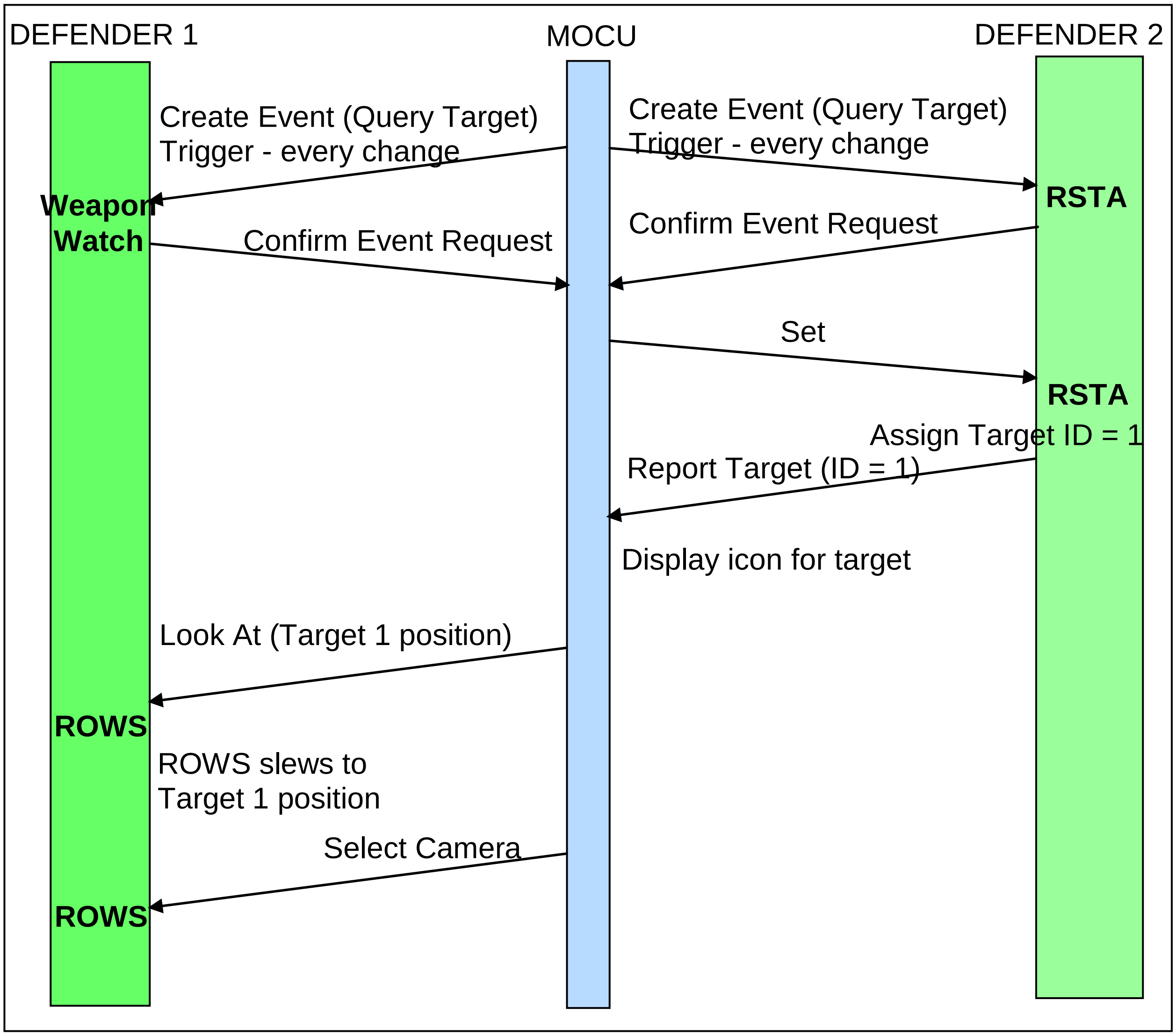


Figure 26. Scenario 1 – OCU Uses Defender 2 to Set Target and Defender 1 to Engage

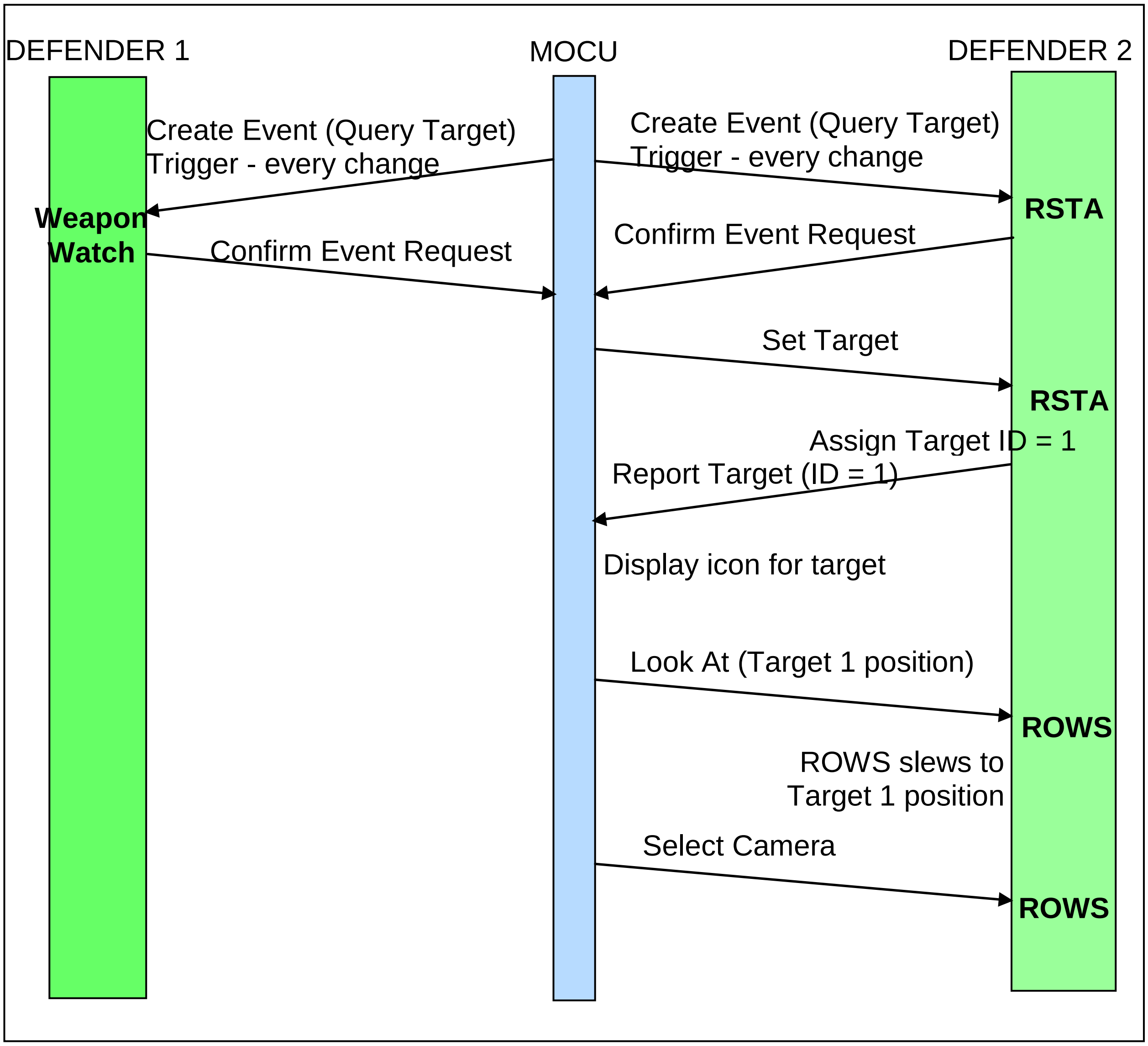


Figure 27. Scenario 2 – OCU Uses Defender 2 to Set Target and Defender 2 to Engage

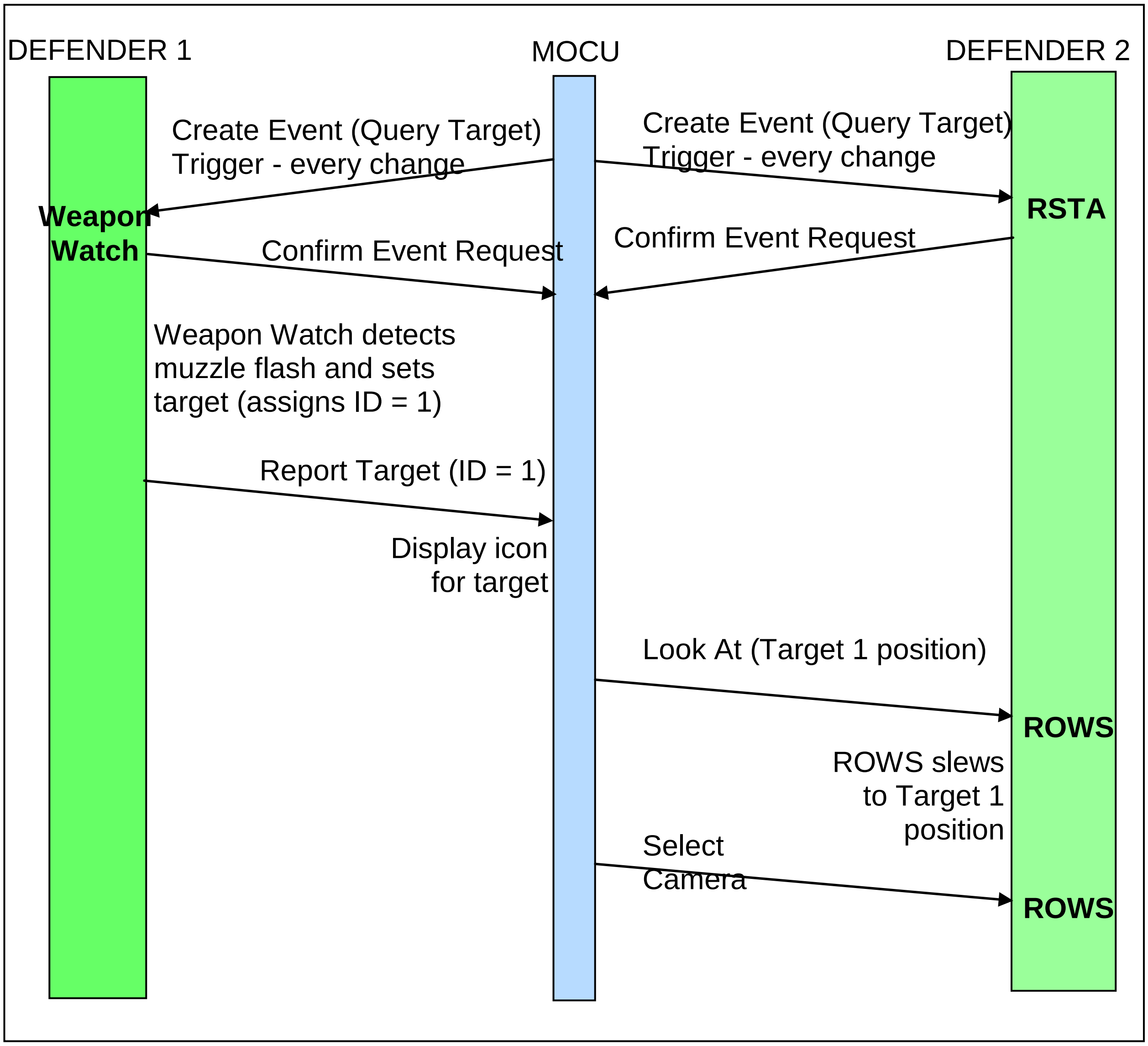


Figure 28. Scenario 3 – Defender 1 Automatically Sets Target, Reports it to the OCU and the OCU Commands Defender 2 to Engage

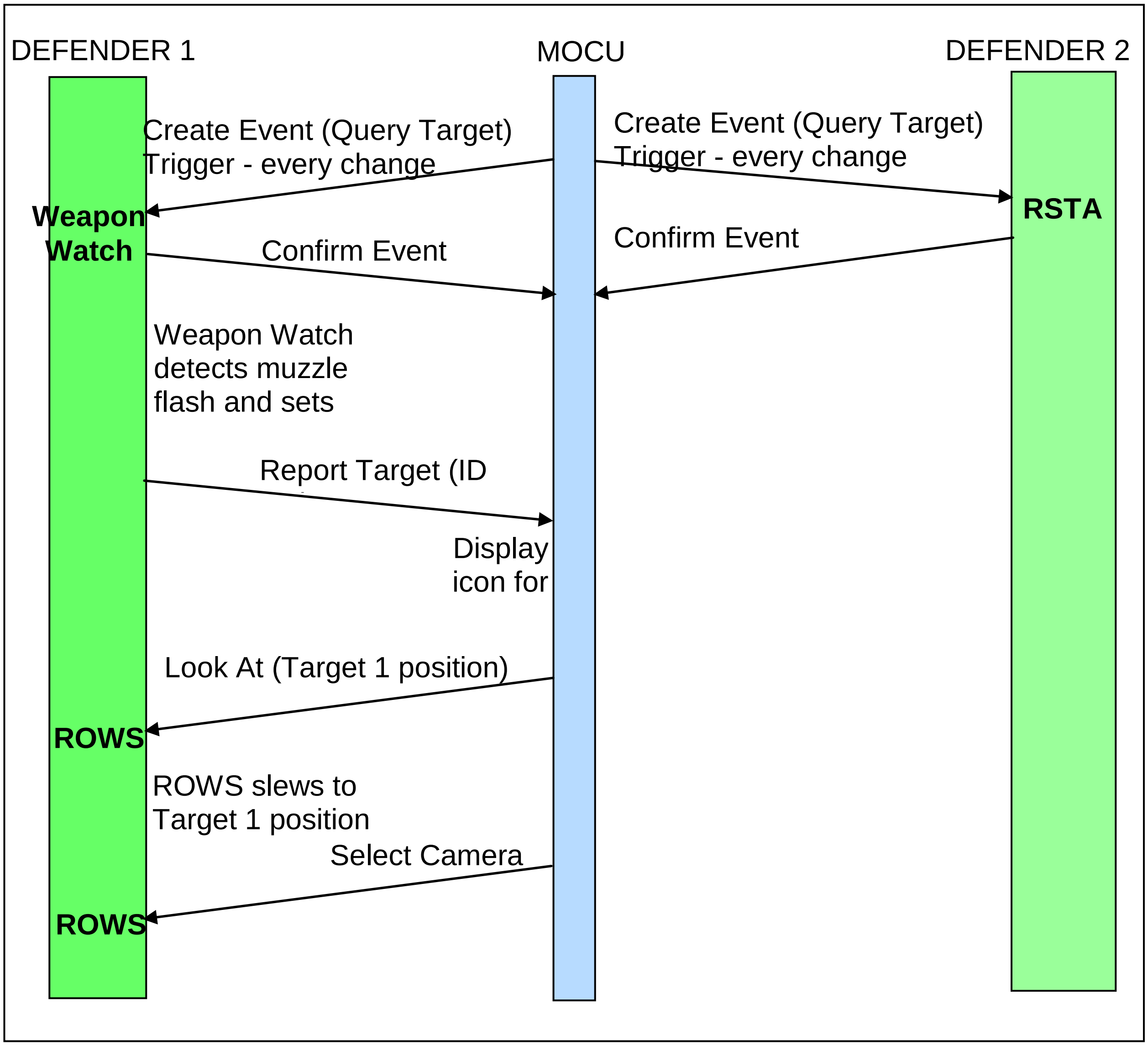


Figure 29. Scenario 4 – Defender 1 Automatically Sets Target, Reports it to the OCU and the OCU Commands Defender 1 to Engage

5.1.2.2.2. Target Detection System - Service ID 80

This service provides target information about detections from sensors like cameras. The TDS is a sensor or group of sensors capable of calculating position of detected targets. The targets can then be classified depending on the capabilities of the Intrusion Detection System (IDS).

The target data is packaged in a message and sent from the sensor to the OCU. The operator, in turn, decides on the action to be taken and can command a different component (e.g. ROWS) on the same or a different subsystem to perform the desired action.

- **Code F500h: Set Target**
  This message is used to set the current visual location or detection as a target. This requires an ACK/NAK from the receiving subsystem. The subsystem that sets the target then assigns it a unique ID (from 1 to 65535). This is used for other subsystems to query for the target information (see Query and Report Target).

- **Code F501h: Query Target List**
  This message will cause the receiving component to respond with a Report Target (F501h). This message is used to query the current list of targets.

- **Code F502h: Report Target List**
  This message is used to report the current list of targets. This message is sent in response to a Query Target List message (F506h). The target list includes the number of active targets followed by the list of target IDs (Table 16).

Table 16. JAUS Byte Field Population for Messages F500…1,2h: (Set,Query,Report) Target List

Field #	Name	Type	Units	Interpretation
1	Number of Targets, N	Unsigned short	N/A	0 – no targets 1-65535: valid number
2	Target 1 ID	Unsigned Short	N/A	0 – reserved 1 to 65535 - unique ID of target
N+1	Target N ID	Unsigned Short	N/A	0 – reserved 1 to 65535 - unique ID of target

- **Code F503h: Query Target**
  This message will cause the receiving component to respond with a Report Target (F501h). This is used to query for information on a specific target (Table 17).

Table 17. JAUS Byte Field Population for Message F503h: Query Target

Field #	Name	Type	Units	Interpretation
1	Presence Vector	Unsigned short	N/A	Refers to Report Target
2	ID	Unsigned Short	N/A	0 – current target 1 to 65535 - unique ID of target

- **Code F504h: Report Target**
  This message is used to report information on a specific target. This message is sent in response to a Query Target message (F500h). The target information includes latitude, longitude, elevation, azimuth, elevation angle, range data, and a time associated with the position information.

  The Target ID is used to tag each unique target (as differentiated by the sensor). This is useful if the sensor has tracking ability. The same target may reappear with a different ID if the sensor determines that it has lost track of the original one.

The triggering of this message may be set-up using the Create Event message with the reports sent on occurrence of a new target (Table 18).

Table 18. JAUS Byte Field Population for Message F504h: Report Target

Field #	Name	Type	Units	Interpretation
1	Presence Vector	Unsigned Short	N/A	Refers to fields 2 - 10
2	ID	Unsigned Short	N/A	0 – reserved 1 to 65535 - unique ID of target
3	Type	Unsigned Short	N/A	0 – reserved 1 – personnel 2 – vehicle 3 – weapon fire 4 – 65535 : reserved
4	Latitude	Integer	Degrees	Scaled Integer Lower Limit = -90 Upper Limit = 90
5	Longitude	Integer	Degrees	Scaled Integer Lower Limit = -180 Upper Limit = 180
6	Altitude	Integer	Meters	Scaled Integer Lower Limit = -10,000 Upper Limit = 35,000
7	Azimuth	Short Integer	Radians	Scaled Integer Lower Limit = $-\pi$ Upper Limit = π
8	Elevation Angle	Short Integer	Radians	Scaled Integer Lower Limit = $-\pi$ Upper Limit = π
9	Range	Integer	Meters	Scaled Integer Lower Limit = -10,000 Upper Limit = 10,000
10	GPS Time Stamp	Unsigned Integer		Bits 0-9: milliseconds, range 0-999 Bits 10-15: Seconds, range 0-59 Bits 16-21: Minutes, range 0-59 Bits 22-26: Hour (24 hour clock), range 0-23 Bits 27-31: Day, range 1-31

- **Code F505h: Clear Target**
  This message will clear the target with a specified ID (Table 19).

Table 19. JAUS Byte Field Population for Message F505h: Clear Target

Field #	Name	Type	Units	Interpretation
1	ID	Unsigned Short	N/A	0 – current target 1 to 65535 - unique ID of target to be cleared

- **Code F506h: Clear Target List**
  This message will clear all active targets.

5.1.2.2.3. Visual Sensor Service (User-Defined).

Once a target has been established and target information is sent back to OCU, the operator can issue a Look At command to a particular component on a subsystem.

Code F510h: Look At

This message is used to point the current camera to a specific location. The location is specified in latitude, longitude and altitude. To differentiate, Set Camera Pose moves the camera using local camera coordinates or rotation rates (Table 20).

Table 20. JAUS Byte Field Population for Message F510h: Look At

Field #	Name	Type	Units	Interpretation
1	Latitude	Integer	Degrees	Scaled Integer Lower Limit = -90 Upper Limit = 90
2	Longitude	Integer	Degrees	Scaled Integer Lower Limit = -180 Upper Limit = 180
3	Altitude	Integer	Meters	Scaled Integer Lower Limit = -10,000 Upper Limit = 35,000

5.1.2.2.4. Teaming

The goal for the JCTE is to show that teams can be formed using the different assets including OCUs, UGVs and UASs. Since all the JAUS commands are initiated from the OCU, it will be assumed that the Team Leader role will be taken by the OCU forming the team. The teaming structure will have the main OCU be the team leader and request membership from (1) secondary OCUs, (2) Base Position Sensor, and (3) all the UGV and UAS. (Figure 30).

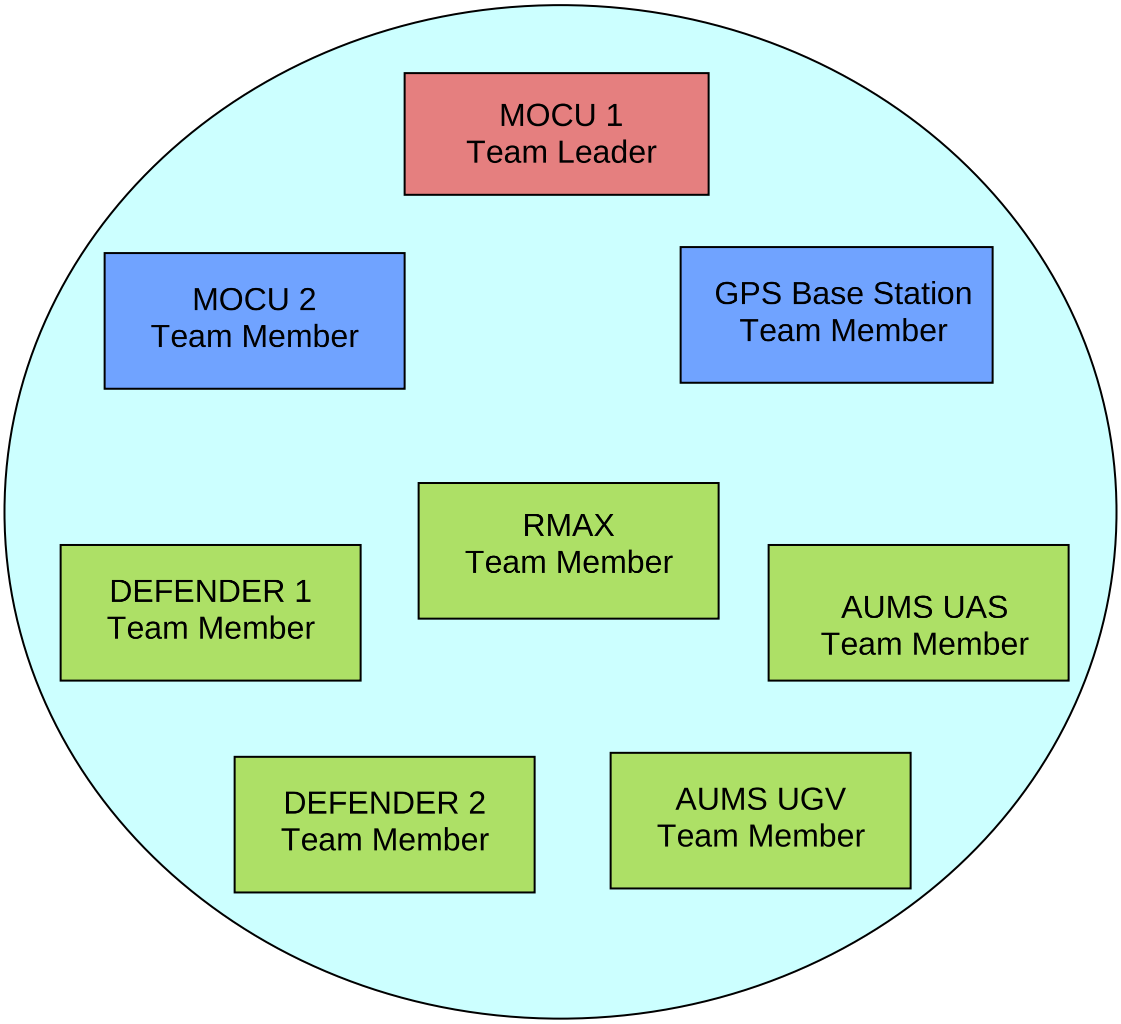


Figure 30. Teaming Structure

- **Code DA00h: Request Team Leadership/Membership**
  This message is used to request that a component join a team either as the leader or as a member. Upon receiving this message, the receiving component must compare the Team Lead ID in the message with its own Source ID. If the IDs match, it should recognize that the request is for it to assume leadership of a team. If the IDs do not match, then it should recognize that the message is a request for it to join a team. If the component supports team lead/member functionality, it can then assume Team Lead or Team Membership. If not, it can report this functionality is not supported. Upon establishment as a team leader, the receiving component will be able to create teams to control directly or to pass messages to from a higher authority. The authority code provided by the requestor is assumed to be that of its direct superior. The component therefore assumes an authority code of one less than the originator. When a component joins a team, it takes note of the authority of the requesting component. If another Team Membership request is received, the authority in the message is compared to the original authority. If the new authority is higher, the component joins the new team. If the authority is equal to or lower than the

one in memory, membership is not accepted. The team designation is the Team Lead's source ID (Table 21).

Table 21. JAUS Byte Field Population for Message DA00h: Request Team Leadership/Membership

Field #	Name	Type	Units	Interpretation
1	Authority Code	Byte	N/A	Authority 0-255
2	Team Lead Subsystem ID	Byte	N/A	Subsystem ID of Leader component
3	Team Lead Node ID	Byte	N/A	Node ID of Leader component
4	Team Lead Component ID	Byte	N/A	Component ID of Leader component
5	Team Lead Instance ID	Byte	N/A	Instance ID of Leader component

- **Code DA01h: Reply Team Leadership/Membership**
  This message is used to notify a requester that it accepts or rejects a team leadership/membership request from that component (Table 22). When Team Leadership or Team Membership is accepted, with a response code of 0, the component will then be able to establish or join a team. It will then generate, pass, or accept team messages. It will also choose to allow or deny peer connections between its subordinate team members (if any) and outside requestors.

  If the component has already established a team of its own, it should not receive another Team Leadership request. If this is the case, the message likely originated from another component with a lower authority than its Team Lead. Any such requests would be responded to with a code of 1, Leadership Not Accepted. For components not supporting team leadership control capability, the response code value of 2 shall be used.

  If the component does not belong to a team, or already belongs to a Team and a Team Membership request arrives from an authority higher than its Team Lead's, it will then join the new Team and respond with a response code of 0. If the component belongs to a Team and a Team Membership request arrives from an authority equal to or lower than its Team Lead's, it will respond with a response code of 1, Membership not accepted. For components not supporting team leadership control capability, the response code value of 2 shall be used.

Table 22. JAUS Byte Field Population for Message DA01h: Reply Team Leadership/Membership

Field #	Name	Type	Units	Interpretation
1	Response Code	Byte	N/A	Bits 0 and 1: 0 = Leadership\Membership accepted 1 = Leadership\Membership not accepted 2 = Leadership\Membership not supported Bits 3-7: Reserved

- **Code DA02h: Release Team Membership**
  This message is used to relinquish team membership of the receiving component (Table 23). This command is accepted only if received from the Team Leader or from a component of higher authority than the one which sent the original Team Membership message.

Table 23. JAUS Byte Field Population for Message DA02h: Release Team Membership

Field #	Name	Type	Units	Interpretation
1	Authority Code	Byte	N/A	Authority 0-255

- **Code DA03h: Add Team Member**
  Once a component has accepted membership within a team, other team members may be made known to the component using this command (Table 24). This command is sent to the component from the team lead. The component is responsible for holding a list of members within its team.

Table 24. JAUS Byte Field Population for Message DA03h: Add Team Member

Field #	Name	Type	Units	Interpretation
1	Member Subsystem ID	Byte	N/A	Subsystem ID of Member component
2	Member Node ID	Byte	N/A	Node ID of Member component
3	Member Component ID	Byte	N/A	Component ID of Member component
4	Member Instance ID	Byte	N/A	Instance ID of Member component

- **Code DA04h: Remove Team Member**
  When components are reassigned to other teams, this message is used to inform the members of the remaining team that the member has left the group (Table 25). This

command is sent to the component from the team lead. The component is responsible to remove this address from the list it holds of members within its team.

Table 25. JAUS Byte Field Population for Message DA04h: Remove Team Member

Field #	Name	Type	Units	Interpretation
1	Member Subsystem ID	Byte	N/A	Subsystem ID of Member component
2	Member Node ID	Byte	N/A	Node ID of Member component
3	Member Component ID	Byte	N/A	Component ID of Member component
4	Member Instance ID	Byte	N/A	Instance ID of Member component

- **Code EA05h: Query Team Membership**
  This message is sent to a component to inquire what team it is assigned to.

- **Code FA05h: Report Team Membership**
  This message is a response to the team membership query (Table 26). It serves to inform the requestor of the designation assigned to that component's Team and Team Leader.

Table 26. JAUS Byte Field Population for Message FA05h: Report Team Membership

Field #	Name	Type	Units	Interpretation
1	Member Subsystem ID	Byte	N/A	Subsystem ID of Team Leader component
2	Member Node ID	Byte	N/A	Node ID of Team Leader component
3	Member Component ID	Byte	N/A	Component ID of Team Leader component
4	Member Instance ID	Byte	N/A	Instance ID of Team Leader component

- **Code DA06h: Request Peer Connection**
  This message is used to request a peer connection between the receiving component and a sending component (Table 27). This message is sent to the component's team leader if one exists. If the leader does exist, this request is accepted or rejected by that component's team lead. When established, the receiving component shall only execute commands from the team lead or peer until the connection is terminated.

Table 27. JAUS Byte Field Population for Message DA06h: Request Peer Connection

Field #	Name	Type	Units	Interpretation
1	Member Subsystem ID	Byte	N/A	Subsystem ID of Team Member component desired
2	Member Node ID	Byte	N/A	Node ID of Team Member component desired
3	Member Component ID	Byte	N/A	Component ID of Team Member component desired
4	Member Instance ID	Byte	N/A	Instance ID of Team Member component desired

- **Code DA07h: Set Peer Connection**
  This message is used to set a peer connection between the receiving component and a sending component (Table 28). It is typically sent by the team lead to both the requesting component and the subordinate team member, letting both know of the grant status. If the team lead denies the request for a peer connection, then only the requestor receives this message, with a connection code of 0. Otherwise, both receiving components receive a message with a sequentially numbered connection code that both associate with the specific peer connection granted. The source ID sent to each component is the source ID of the peer which it will establish a link with.

Table 28. JAUS Byte Field Population for Message DA07h: Set Peer Connection

Field #	Name	Type	Units	Interpretation
1	Authority Code	Byte	N/A	Authority 0-255
2	Connection Code	Byte	N/A	Connection 0-255
3	Team Member	4-Bytes	N/A	Source ID of Peer

- **Code DA08h: Terminate Peer Connection**
  This message is used to terminate a connection that has been established between two components using a peer connection (Table 29).

Table 29. JAUS Byte Field Population for Message DA08h: Terminate Peer Connection

Field #	Name	Type	Units	Interpretation
1	Authority Code	Byte	N/A	Authority 0-255
2	Connection Code	Byte	N/A	Connection 0-255

5.1.2.3. AUMS Host UGV.

The AUMS Host UGV is a tele-operated HMMWV. Since its primary mission is a host for the AUMS Base and AUMS UAS, only vehicle control and position messages are implemented from the JAUS 3.2 standard [2]. A subset of the JAUS OPC 2.75 [3] messages is implemented so that the vehicle will operate with MOCU. All of the experiment Teaming messages (5.1.2.2.4) are implement with the exception of the peer connection messages as they will not be needed for this experiment.

The AUMS Host UGV supports messages for the following services:

1. Node Manager
2. Primitive Driver
3. Global Pose Sensor
4. Velocity State Sensor
5. Teaming

1. Node Manager – Refer to JAUS R.A. 3.2 and OPC 2.75 for dynamic discovery messages.
2. Primitive Driver – Refer to JAUS R.A. 3.2

- Set Wrench Effort
    - Propulsive Linear Effort X – Sets the vehicle forward or reverse velocity from 0-100%. Direction is set via the Discrete Devices message.
    - Propulsive Rotational Effort Z – Sets the vehicle steering from -100 – 100 %.
    - Resistive Linear Effort X – Sets the vehicle braking from 0 – 100%
    - Query Wrench Effort
    - Report Wrench Effort – Vehicle will respond to a Query Wrench Effort with the following data:
    - Propulsive Linear Effort X – Reports the commanded forward or reverse velocity setting of the vehicle from 0-100%.
    - Propulsive Rotational Effort Z – Reports the commanded steering setting of the vehicle from -100 – 100 %.
    - Resistive Linear Effort X – Reports the commanded vehicle braking from 0 – 100%
- Set Discrete Devices
    - Main Propulsion – Bit 0 turns the vehicle engine on and off.
        - 0 – turns the engine off
        - 1 – turns the engine on.
    - Parking Brake, Horn – Bit 0 enables and disables the HMMWV parking brake.
        - 0 – enables the parking brake
        - 1 – disables the parking brake.
    - Gear – Controls the vehicle transmission. There are only three valid inputs and transmission states:
        - 127 – Drive
        - 128 – Neutral, the default state. The vehicle should be put into neutral whenever the parking brake is set.
        - 129 – Reverse
- Query Discrete Devices
- Report Discrete Devices – Vehicle will respond to a Query Discrete Devices with the following data:

    - Main Propulsion – Reports commanded engine state using Bit 0 as defined above.
    - Parking Brake, Horn – Reports the commanded state of the vehicle parking brake in Bit 0 as defined above.
    - Gear – Reports the commanded position of the vehicle transmission as defined above.

3. Global Pose Sensor – Refer to JAUS R.A. 3.2
    - Query Global Pose
    - Report Global Pose – Vehicle will respond to a Query Global Pose with the following data:
        - Current vehicle yaw (heading)
        - Current vehicle pitch
        - Current vehicle roll

4. Velocity State Sensor – refer to JAUS R.A. 3.2
    - Query Velocity State
    - Report Velocity State – Vehicle will respond to a Query Global Pose with the following data:
        - Velocity X – Current velocity along the vehicle X-axis (along heading).

5. Teaming – See Teaming message definitions in 5.1.2.2.4 above.
    - Request Team Leadership/Membership
    - Reply Team Leadership/Membership
    - Release Team Leadership/Membership
    - Add Team Member
    - Remove Team Member
    - Query Team Membership
    - Report Team Membership

5.1.2.4. AUMS Base

The JAUS interface for AUMS base includes standard messages from RA 3.2, OPC 2.75 and one user-defined component for basic operation.

The AUMS Host UGV supports messages for the following services:

1. Node Manager
2. Global Pose Sensor
3. Velocity State Sensor
4. Visual Sensor
5. Teaming
6. Fuel Pump (User Defined)

1. Node Manager – Refer to JAUS R.A. 3.2 and OPC 2.75 for dynamic discovery messages.

2. Global Pose Sensor – Refer to JAUS R.A. 3.2
    - Query Global Pose

- Report Global Pose – Vehicle will respond to a Query Global Pose with the following data:
    - Current vehicle yaw (heading)
    - Current vehicle pitch
    - Current vehicle roll

3. Velocity State Sensor – Refer to JAUS R.A. 3.2
    - Query Velocity State
    - Report Velocity State – Vehicle will respond to a Query Global Pose with the following data:
        - Velocity X – Current velocity along the vehicle X-axis (along heading).

4. Visual Sensor – Refer to JAUS R.A. 3.2
    - Query Image
    - Report Image – System will respond to a Query Image with a Report Image
    -

5. Teaming – See Teaming message definitions in 5.1.2.2.4.
    - Request Team Leadership/Membership
    - Reply Team Leadership/Membership
    - Release Team Leadership/Membership
    - Add Team Member
    - Remove Team Member
    - Query Team Membership
    - Report Team Membership

6. Fuel Pump
    - Query Component Status
    - Report Component Status – Fuel Pump Component will respond to a Query Component Status with the following data:
        - UAS centering status,
        - UAS capture status
        - Inject fuel pod status
        - Fuel amount
        - Defuel amount
        - Release fuel pod status
    - Set Component Status – Fuel Pump Component will execute the following commands within this message:
        - UAS centering,
        - UAS capture
        - Inject fuel pod
        - Fuel
        - Defuel
        - Release fuel pod

5.1.2.5. AUMS UAS

The JAUS interface for AUMS UAS involves standard messages from RA 3.2 and OPC 2.75. The AUMS UAS supports messages for the following services:

1. Node Manager
2. Global Pose Sensor
3. Velocity State Sensor
4. Visual Sensor
5. Global Vector Driver
6. Global Waypoint Driver
7. Teaming

1. Node Manager – Refer to JAUS R.A. 3.2 and OPC 2.75 for dynamic discovery messages.

2. Global Pose Sensor – Refer to JAUS R.A. 3.2

- Query Global Pose
- Report Global Pose – Vehicle will respond to a Query Global Pose with the following data:
    - Current vehicle yaw (heading)
    - Current vehicle pitch
    - Current vehicle roll

3. Velocity State Sensor – Refer to JAUS R.A. 3.2

- Query Velocity State
- Report Velocity State – Vehicle will respond to a Query Global Pose with the following data:
    - Velocity X – Current velocity along the vehicle X-axis (along heading).

4. Visual Sensor – Refer to JAUS R.A. 3.2

- Query Image
- Report Image – System will respond to a Query Image with a Report Image

5. Global Vector Driver – Refer to JAUS R.A. 3.2

6. Global Waypoint Driver – Refer to JAUS R.A. 3.2 and these two messages for vertical takeoff and landing

- Launch Vehicle
    - Yaw – Direction UAS to face at the climb out altitude
    - Altitude – Height UAS to climb out to
- Land Vehicle
    - Altitude – Landing altitude
    - Latitude and Longitude – Landing position

7. Teaming – See Teaming message definitions in section 5.1.2.2.4above.

- Request Team Leadership/Membership
- Reply Team Leadership/Membership
- Release Team Leadership/Membership

- Add Team Member
- Remove Team Member
- Query Team Membership
- Report Team Membership

5.2. Integration Sessions

5.2.1. First Integration Session

The goals of the first integration session were (1) verify the IP scheme and check the network configuration using a wired connection, (2) verify the vehicle/subsystem discovery process using MOCU, (3) test and verify critical feedback including position, velocity and status, and (4) test video and measure bandwidth utilization.

All these checks were performed with the Comm Repeater remote side wired to the base side eliminating all wireless links. Also, except for Defender 1 and 2, vehicle simulations were used since no actual vehicle hardware was present.

There were no issues in achieving the first goal. All the IP addresses assigned according to the IDD were tested and no conflicts arose. However, since not all of the actual hardware to be used was present, not all the IP addresses were verified.

The MOCU was successful in discovering the following vehicles/systems: Defender 1, Defender 2, AUMS UAS, AUMS, and RMAX. All the above systems sent back valid position, velocity and status data to the MOCU.

For video, there were two types of video data used, JAUS and non JAUS. Defender 1 and 2 sent back JAUS UDP video at 320x240 resolution at 15 Hz. When benchmarked, the bandwidth used was 0.8 to 1 Mbps. The AUMS UAS used an Axis video server which sent Internet Protocol Suite (TCP/IP) video at 320 x 240 resolution ant 15 Hz. The bandwidth used was 0.5 to 1.5 Mbps. It was noted that a potential issue could arise that bad communications would result in extra bandwidth used because of data retries, a characteristic of TCP/IP. This was never tested since our connection was wired. The same Axis video data was used by the AUMS Host UGV.

5.2.2. Second Integration Session

The second integration was an extension of the first integration session in that this session addressed unfinished tasks in the first and the wireless links were to be introduced. The main goals of the second integration were (1) verify network configuration using both wired and wireless connection, (2) using the MOCU, verify discovery, status and feedback of vehicles, (3) test the LMS, (4) test Targeting and Pointing, and (5) test auto tracking antenna for Communications Repeater L-band link at short and long ranges.

Unlike the first integration session, the goal was to use actual vehicle hardware. Unfortunately, the AUMS and AUMS UAS were not present. This left Defender 1 and 2, the AUMS Host UGV, and the RMAX as available vehicles.

In verifying the network, all vehicles, MOCUs and the Comm Repeater were first connected wired as in the first integration session. Once the wired network was verified, the 2.4 GHz wireless Ethernet link was added. The vehicles used wireless client radios to talk to the access point radio on the Comm Repeater pod. In establishing a good communications link to the ground vehicles, the only issue was antenna placement on the AUMS Host UGV which was critical.

Using the MOCU, each vehicle was separately discovered and tested for control and feedback. This meant having only one of the vehicles on at any time. There were no apparent issues. However, when the vehicles were turned on all at the same time, there were problems in the discovery process between the DEFENDER vehicles and the AUMS Host UGV. There was constant rediscovery that would cause the AUMS Host UGV computer to drop out.

In testing the LMS, simulated position for DEFENDER 1 and 2 and the AUMS Host UGV were used. The LMS was put in a mode where it automatically generated waypoints for the RMAX. The simulated vehicles would then be repositioned to make the LMS recalculate a new waypoint.

Targeting and Pointing were tested using MOCU and DEFENDER 1 and 2. Targeting was done by having DEFENDER 2 determine a target using its laser range finder. The target would then appear as an icon on the map on MOCU. Pointing was performed by using the MOCU to send the target position information DEFENDER 1, commanding the selected pan/tilt camera to point at that location. Results showed that the pointing was within 3 degrees of the actual target. A larger error was noticed in the elevation of the target with respect to where DEFENDER 2's camera pointed.

The last goal was to test the tracking antenna. Initially, the tracking antenna, UCR, and the MOCU were positioned approximately 0.5 miles away from the RMAX. The RMAX was flown manually at an altitude of 50 meters. The RMAX base would send the RMAX position back to the tracking antenna controller for it to track. This meant that good communication had to be established first before the antenna could start tracking. The antenna tracked accurately whenever good communication was maintained. The long range test was not performed due to lack of time.

5.2.3. Third Integration Session

The third scheduled integration session was the final stage in the integration process and concluded with the actual experiment and technology demonstration. This was the first time that all the individual system hardware (MOCU, DEFENDER, RMAX, UCR, AUMS UAS, AUMS, and AUMS Host UGV) were together for total system integration. Factors that contributed to the delay in total integration were (1) change in the platform used for the AUMS UAS, (2) upgrading the RMAX WePilot hardware, and (3) late delivery of the tracking antenna.

Up to this point using the MOCU, the individual systems were tested and verified separately, however, only limited testing was done with multiple systems. The first few days were spent performing preliminary system checks including (1) JAUS IDD [1] compliance, (2) frequency management, (3) CR tracking antenna configuration, and (4) bandwidth utilization management.

Due to limited time, compliance was focused on the essential areas, including vehicle command and control, position and velocity feedback, and vehicle status feedback. Command and Control included vehicle mobility and payload control. Compliance in (1) the MOCU control of the LMS, and (2) Teaming, was left for future work.

A big challenge was frequency management. When the RMAX WePilot controller was upgraded, the radio it used for the autopilot changed to a 2.4GHz frequency-hopping spread spectrum (FHSS) radio. There were two other radios on 2.4GHz, the Esteem radios, and the AUMS UAS video radio. All three radios were assigned channels separated from each other enough to eliminate interference. However, during the ground and air tests, the RMAX radio was showing a high percentage of bit errors. The solution was to move the RMAX autopilot link to 900-MHz. This change put it in conflict with the RC control radio for the AUMS UAS. The lack of time and radio options resulted in the compromise of not operating the two helicopters simultaneously.

The next check was the tracking antenna location and configuration. The experiment was set-up for long range remote operation of 4 miles. There was a limitation in allowed ceiling of 500 meters for flying the RMAX during the demonstration. Radio link tests were conducted at altitudes of 400 meters and 500 meters. The initial location for the tracking antenna was outside the north end of the Test Range 3 building. The L-band link was marginal at best. To counter this, the tracking antenna was relocated 300 meters northwest of the original spot. This allowed the antenna to get around the tree line that was affecting the L-band signal. In order to allow the operators to still conduct the demonstration from inside the building, the tracking antenna was bridged to the building using a second Esteem radio link. This was operating on a different channel as that of the Esteem on the Comm Repeater. During the radio link tests at 500 meters, the link improved significantly. The only other factor that compromised radio link quality was shielding due to antenna placement on the helicopter.

The problems with the frequency deconfliction and the tracking antenna configuration pushed back the schedule and left no time for actual experimentation and data collection before the demonstration. Even the final system configuration would not allow all the individual systems to operate simultaneously, thus making it less than ideal for experimentation and data collection.

The demonstration was divided into three parts that highlighted several key functional areas of the JCTE. Part 1 demonstrated the long range remote operation of DEFENDER 1, DEFENDER 2, and the AUMS Host UGV. The operation included targeting and pointing. Part 2 was the autonomous mission capabilities of the AUMS UAS. The last part dealt with the autonomous refueling of the AUMS.

6. JCTE DEMONSTRATION DESCRIPTION AND RESULTS

6.1. General

JCTE partners conducted a successful Proof of Concept Technology Demonstration on 9 October 2008 at Tyndall AFB's Silver Flag Exercise Site. JGRE Partners, including the AFRL/RXQ, ARMDEC SED, and SPAWAR Systems Center San Diego, capped a nine-day collaborative technologies experimentation window with a demonstration to invited representatives from DoD organizations including the Air Force's Security Forces Center, Air Combat Command (ACC/A7S), the Army's TRADOC Capabilities Manager for Future Combat Systems, and the Robotic Systems JPO.

Visitors observed networked air and ground unmanned systems operations performing Base Defense missions and exhibited autonomous, semi-autonomous and collaborative behaviors in remote operations over an extended communications link of nearly five miles. A briefing was included on supporting simulations that further documented the value added of JCTE technologies in Base Defense mission scenarios.

The JGRE funded JCTE project focus was to provide enhanced, networked technology enablers and capabilities to Air Force Security Forces and other DoD users in executing Base Defense missions. JCTE demonstrated increased mission capabilities resulting from integrated Collaborative Technology enhancements while operating under the Navy's interoperable MOCU-JAUS command and control system and featured networked targeting and engagements, beyond line of sight communications link management, and autonomous UAS launch, ISR mission execution, landing, refueling, and re-launch. Another JCTE focus was to provide warfighters additional capabilities without significant increases to operator workloads. JCTE partners are proceeding with planning to further refine and demonstrate these capabilities in an Air Force sponsored Warfighting Experiment in FY09.

6.2. Demonstration Layout

Silver Flag was an ideal site for experimentation due to its location within five miles of AFRL's Robotics JCTE Research Team. Several factors made this site ideal including: the availability of a full sized airfield replicating the anticipated JCTE operational environment and the availability of restricted airspace and ground space for full utilization of ground and air robotics platforms. Figure 31 illustrates the overall operational demonstration area.

Figure 31. Site Location

The airspace at Silver Flag provided the opportunity for JCTE partners to operate the ground and air robotic technologies remotely over a 4.5mile line of sight link from the AFRL Robotics Compound to the Exercise Airfield site. Figure 32shows a zoomed in aerial perspective of the Silver Flag Site.

Figure 32. Silver Flag Site Location

6.3. Experiment Item Description

6.3.1. Patrol/Engagement Platform

The function of the engagement platform is to perform the challenge, response, delay/denial and neutralization function of the automated perimeter security (APS) system. The engagement platform must travel at high speeds (up to 35 mph), negotiate wooded areas, be “zero-radius” turn capable, and travel through rough terrain (at reduced speeds). Rough terrain includes steep slopes up to ±20 degrees side slope, tall grass areas with soft bedding, shallow swamps and ditches (up to 12-in deep), and areas with trees 75-in or more apart. The platform must have a minimum mission endurance of 6 hours.

The engagement platform selected for the APS mission is the “Defender” developed by the Robotics Development and Research Team at AFRL. This vehicle is the same platform utilized successfully for AFRL limited user experiments in 2005 at the Francis E. Warren AFB and in 2007 at the Kirtland Munitions Storage Complex. The base vehicle to the Defender is the Land Tamer II 6X6, a diesel powered vehicle developed by PFM Manufacturing. The vehicle comes equipped with a 17 gallon fuel tank, giving an estimated 17-hour run time at fifty percent power. This vehicle was selected for its overall low cost, skid steering capability and electric over hydraulic actuator system currently used to control the vehicle. The engagement platform has a color camera system for video feedback as the primary driving reference. It has a strobe/flashing light system, a speaker microphone system for subject interaction, and lethal weapon systems

incorporated into the base platform. The lethal weapon for this system is either the M-16A2 rifle or M-240/M249 machine gun. The lethal weapon is mounted in and controlled with a TRC XROWS^{tm}. Figure 33depicts the base vehicle for the Defender system.

Figure 33. Defender Engagement Systems

6.3.2. RMAX UAS

The UAS platform is a COTS Yamaha RMAX Model L-17 rotary wing UAS (Figure 34). The RMAX UAS has water-cooled, 2-stroke, horizontally opposed 2-cylinder 246cc, 2 Stroke cycle gas engine, 3,115mm (10.21ft) main rotor diameter, and a 30Kg (66lb) payload capacity and is typically used commercially for agricultural crop-dusting. The RMAX has two forms of control. The primary control is the native Yamaha RC remote control. The primary control works over a 72MHz radio link and provides direct pilot control of the aircraft throttle, main rotor pitch, and tail rotor pitch. This direct pilot control is stabilized by the native Yamaha rate gyro system onboard the aircraft. This control scheme is common to most radio control rotary wing models.

Figure 34. RMAX Rotary Wing UAS

The secondary control is through the COTS WePilot autopilot system. The WePilot system consists of an autopilot control unit, radio link, and a ground control station. The autopilot control unit is a microprocessor based system that receives input from onboard sensors (GPS, rate gyro, and engine rpm) and directs the Yamaha flight controller based on ground control station commands or stored waypoint paths.

The autopilot control unit is installed onboard the aircraft and is connected directly to the native Yamaha flight controller. When activated, the RMAX aircraft receives flight control signals from the WePilot autopilot control system rather than the Yamaha remote control; however primary control is always maintained by the Yamaha remote control over the 72-MHz radio link. The Yamaha remote control has a mode switch that allows the pilot to select which source has flight control command authority, either the Yamaha remote control or the WePilot system. The Yamaha remote control is always in communication with the aircraft over the 72-MHz radio link and the pilot can switch off reception of the WePilot commands at any time.

The WePilot uses a single 2.4 GHz, 100mW radio link and single antenna pair for both data and video communications. The WePilot ground control station consists of a custom interface console and a laptop computer. The console provides joysticks, sliders, and buttons for UAS flight surface, throttle, and payload controls. The selected payload video is displayed on a small video screen in the console case lid. The laptop computer is used to program, execute, and

monitor flight operations and waypoint path performance. The laptop computer displays a map of the operating area and aircraft flight parameters.

6.3.3. AUMS

The AUMS is designed to provide current and future military programs with the capability to automatically launch, recover, refuel, and re-launch a small VTOL UAS (Figure 35). This capability will solve a problem that hinders the adoption and expansion of these platforms in the battlefield. Their utility is greatly diminished by their limited payload and duration.

Since UASs require frequent refueling, they spend less time in the field of operation, which reduces their effectiveness in many military environments. The effective payload and duration of small UASs can be increased by moving the support base closer to the area of operation, but this increases risk to personnel. If the refueling and rearming operations can be performed autonomously, then the support base can be transported closer to the area of operation without increasing risk to personnel. The AUMS development effort is divided into three major phases: 1) Launch and Recovery; 2) Refueling; and 3) Landing.

Figure 35. Autonomous UAS Mission System

6.3.4. MOCU Command and Control System

The function of the APS command and control system is to operate the platforms through the use of a single laptop or desktop computer. This is possible through the use of the JAUS specification and the Navy SPAWAR-developed MOCU allowing for a net-centric approach. The JAUS protocol also provides seamless integration between the various robotic platforms and their payloads on the network. Benefits of using the JAUS common architecture include: common control commands, single RF network, sensor fusion, and the ability to pass control between operators. A single type OCU allows for common control commands for all platforms. A manual Emergency-Stop is mounted on the rear panel of the ground vehicles.

The command and control system consists of an OCU computer, the network communication gear, the radio frequency transceivers, and the antenna network. The computer is a standard personal computer with sufficient processing power, memory, and features to perform the OCU functions. The computer uses a Windows based operating system loaded with the JAUS control system software. The operator’s primary display is through the monitor and switches depending on which platform is currently being operated. The network communication uses a wireless Ethernet network. Each platform has a radio unit for both RF communication and network routing duties. The C2ISR sensors are patched through a 100T Ethernet hub to the OCU. A network of antennas are used throughout the operational area to provide continuous coverage.

6.4. Method of Experiment

6.4.1. Approach and Case Definition

Performance capabilities of the APS system and in particular the Defender platform were the subjects of a USAF AFRL experiment at the Kirtland Storage Area in 2007. The 2008 JCTE experiment utilized the Defender as representative ground robotics Security Force support platforms, but the focus of the experiment was to evaluate the utility of incorporating additional communications and unmanned aerial vehicles with embedded collaborative capabilities to enhance overall Base Defense capabilities. These enhanced capabilities were demonstrated through a series of mission scenarios designed to replicate events and responses normally associated with perimeter defense activities. Primary supervision and operator responsibilities remained with JCTE personnel. The following scenarios were demonstrated:

- APS (Figure 36) – Reconnaissance and Patrolling, Respond to Hostile Threats, JCTE Scenarios: CASE 1: Base Case – Perimeter Defense
    - Current cooperative technologies, range 1-2-Km LOS
- CASE 2: Base Case – Recon/Patrolling
    - Current cooperation technologies with communications relay, extended range
- CASE 3: Base Case – Respond to Hostile Threats
    - Current cooperative technologies, range 1-2-Km LOS

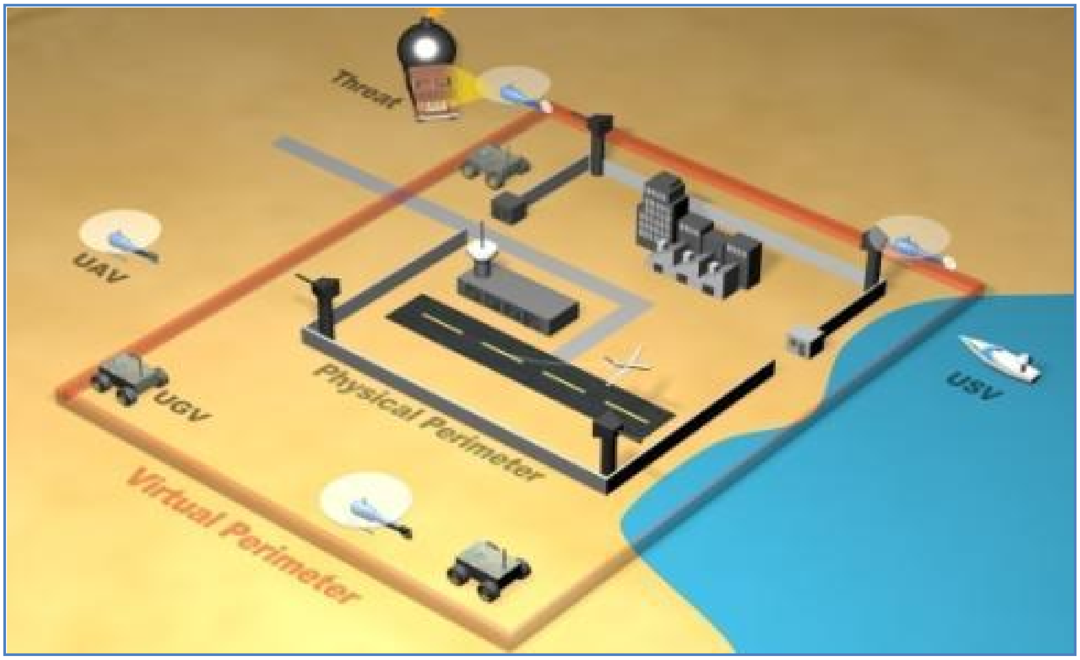


Figure 36. APS Scenario Example

APS – Extended Perimeter Defense with added Collaborative capabilities, JCTE Scenarios:

- CASE 4: Test Case – Recon/Patrolling
  - Collaborative technologies, extended range with communications relay and Link Management System
- CASE 5: Test Case – Recon/Patrolling
  - Collaborative technologies, extended duration using the Autonomous UAS Mission System (AUMS), 30-min flight time/ in a 4-hour mission scenario (actual mission time will be closer to one hour)
- CASE 6: Test Case – Recon/Patrolling
  - Collaborative technologies, extended range with communications relay and LMS, extended duration with AUMS 30-min flight time/in a 4-hour mission scenario (actual mission time will be closer to one hour).
- CASE 7: Test Case – Respond to Hostile Threats
  - Collaborative technologies, target location passing using LMS and pointing/cueing. Unmanned systems will be utilized to induce hostile target delay and denial. Targets will be neutralized using simulated lethal and/or non-lethal means to gain insights into experimental gains in efficiency realized through collaborative targeting/cueing.
- CASE 8: Test Case – Recon/Patrolling & Respond to Hostile Threats
  - Collaborative technologies with future capabilities (Simulations Only)

6.4.2. Conclusions.

The optimistic theories that collaborative robotics are able to provide advantages and efficiencies to the warfighter were positively supported by the results of the JCTE. A fully operational UMS

with the mission to protect people and equipment at a site, be it an airfield or other location, has been shown to be not only feasible but effective while reducing risk to US lives. Properly executed collaborative robotics creates streamlined performance of the mission and reduces the workload on the warfighter.

The LMS is currently not at the functional level that will be required to field such a system to the warfighter. Frequency management and link propagation using a device like the UCR needs further improvement and solidified techniques to provide the warfighter the reliability that is required in today's theater of operations. Antenna and tracking system designs will need to be chosen and optimized for the UAS and UGV to be used.

As configured, the system is vulnerable to jamming, adverse weather, and conceivably even hostile takeover by technically advanced adversaries. Adverse weather is a problem for the two UAS used in the experiment and the choice of specific UASs for the roles envisioned would have to take into account the need for 24 hour a day, seven day a week coverage. The selection criteria for such a system must also take into account operations in high winds, rain, or heavy snow and icing conditions for many Contiguous United States (CONUS) and Outside Contiguous United States (OCONUS) locations. Selection of fieldable UAS platforms should consider a UAS with severe weather capabilities in order to provide 24/7 coverage.

6.4.3. Recommendations.

Improvement that should be pursued in future efforts should include addressing frequency management issues, more fully implementing teaming to increase collaboration, and increasing the operational reliability and capability of the unmanned systems. Implementing these improvements should allow greater capability while simultaneously working to further reduce operator workload. Specific recommendations are:

- Conduct research to improve the performance of the S-band link(s). S-band link performance is greatly affected by vehicle dynamics and antenna types. Typically omni-directional antennas are used on the ground vehicles. These antennas have toroidal patterns that are optimum in the horizontal plane but roll-off significantly with increase in elevation. When using an airborne employed communication repeater node such as the UCR, the air vehicle might be at a relatively high elevation (aka look-up) angle with respect to one or more ground vehicles. Typically, the higher the look-up angle, the lower the signal strength. In addition, the air vehicle will experience changes in attitude that also attributes to scintillation in signal strength. These factors can be mitigated through antenna optimization. Directional or beam steering would contribute greatly to an increase in S-band link performance while adding some additional system complexity.
- Automate setup and configuration of the tracking antenna system. The effectiveness of the tracking antenna system used with the UCR is highly dependent on accurate North alignment, position location, and leveling. At present this setup and configuration is done manually. It is believed that this setup can be automated, thus reducing the overall operator workload associated with utilization of this equipment.
- Conduct testing to validate operation of the UMS at extended ranges (out to 50 miles) using the UCR. Theoretically the UCR with tracking antenna system should support BLOS UMS operations out to 50 miles. Range and terrain limitations along the Gulf Coast inhibited additional testing of the UCR at ranges longer than 15 miles.

- Incorporate the antenna patterns into the path loss equations used by the LMS to achieve a more realistic generation of the regions for expected effective communication links.
- The LMS should be modified to allow the operator to select between an optimum position mode and an acceptable region flight path mode for the UAS type being used, either fixed wing or rotary wing. Other possible operator inputs could allow for real-time adjustment of configuration parameters like antenna gains or transmit powers.
- Dynamic antenna models should be included in the LMS. These models would account for variations in antenna patterns based on vehicle attitude and antenna orientation. Corresponding gains would then be computed by the LMS to account for roll-off in signal strength as a function of elevation look-up angles between a UGV and a UAS carrying a communications payload such as demonstrated in the JCTE effort.
- DTED should be incorporated into the LMS. The LMS would be modified to utilize DTED information to assist in determining the RF propagation shading due to terrain.
- Additional testing should to be performed on the LMS acceptable region profile using a fixed wing or rotary wing UAS to completely validate LMS performance in these modes. The acceptable region for communication is used as the basis for generating the waypoints that create a flight path for the UAS. Following the LMS generated flight path a UAS will remain within the computed expected region for acceptable communication supporting UMS networked operations.
- Incorporate additional path loss algorithms suited for cellular and WiFi communication waveforms and frequencies into the LMS.
- Add a real time operator interface for modification of system parameters such as adjusting the fade margin.
- Test the effectiveness of the LMS with the UCR carried by a fixed wing platform.

7. REFERENCES

1. JAUS Interface Design Document. *Internal Document.*

2. JAUS: The Joint Architecture for Unmanned Systems: Reference Architecture Specification. Volume II, Part 1: Architecture Framework Version 3.3. June 22, 2007. Available from http://www.openjaus.com/products/openjaus33/79-jaus-documents. Accessed 2 May 2013

3. OPC 2.75 Experiment. *Internal Document.*

4. Defender Interface Design Document, v7. *Internal Document.*

LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS

ACC	Air Combat Command
ACK/NAK	acknowledged/not acknowledged
AFB	Air Force Base
AFCESA	Air Force Civil Engineering Support Agency
AFRL	Air Force Research Laboratory
AFRL/RXQ	AFRL Materials and Manufacturing Directorate, Airbase Technologies Division
AGL	above ground level
AL	aluminum
AMRDEC SED	Army Aviation and Missile Research, Development and Engineering Center, ,Software Engineering Directorate
AP	access points
AP1	root AP
AP2	static repeater AP
AP3	dynamic repeater AP
API	application programming interface
APS	automated perimeter security
ATC	air traffic control
ASW	antisubmarine warfare
AUMS	Autonomous UAS Mission System
BLOS	beyond line of sight
C2	command & control
C2ISR	command, control, intelligence, surveillance, & reconnaissance
C4ISR	command, control, communications, computers, intelligence, surveillance, & reconnaissance
CAFC	computer-aided fire control
CCF	collaborative communication cootprint
CCT	cloud cap technology
CEE	Collaborative Engagement Experiment
Comm(s)	communication(s)
CONOPS	concepts of operation
CONUS	continental United States
COP	common operational picture
COTS	commercial off- the-shelf
CR	comm repeater
CR-B	comm repeater base
CR-R	comm repeater remote
DARPA	Defence Advanced Research Projects Agency
DGPS	differential global positioning system
DoD	Department of Defense
DTED	digital terrain elevation data
ECEF	earth centered earth fixed
ENU	East-North-up
EO	electro-optical

EOD	explosive ordnance disposal
FHSS	frequency-hopping spread spectrum
GCS	ground control station
GST	guided systems technology
GPS	global positioning satellite
GUI	graphical user interface
GWD	global waypoint driver
HMMWV	high mobility multipurpose wheeled vehicle
I/O	input(s)/output(s)
ID	identification
IDD	Interface Design Document
IED	improvised explosive device
IMU	inertial measurement init
IP	internet protocol
ISR	intelligence, surveillance, and reconnaissance
JAUS	Joint Architecture for Unmanned Systems
JAUS RA v3.2	JAUS Reference Architecture version 3.2
JCAs	Joint Capability Areas
JCTE	Joint Collaborative Technology Experiment
JGRE	Joint Ground Robotics Enterprise
JRP	Joint Robotics Program
KUMSC	Kirtland Underground Storage Complex
LADF	lift-augmented ducted fan
LCS	Littoral Combat Ship
LiIon	lithium ion
LMS	link management system
LOS	line of sight
MAC ID	media access control identifier
MAV	Micro Air Vehicle
mi	miles
MDARS	Mobile Detection Assessment and Response System
MIW	mine warfare
MOCU	Multi-Robot Operator Control Unit
MSL	mean sea level
OAV	Organic Air Vehicle
OCONUS	outside the continental United States
OCU	operator control unit
OEF	Operation Enduring Freedom
OIF	Operation Iraqi Freedom
OPC	OCU and Payload Committee
PCU	power control unit
R/C	radio control
R&D	research and development
RF	radio frequency
ROWS	Remote Operated Weapons System
RSTA	reconnaissance, surveillance, and target acquisition

Rx	receiver
SAE	Society of Automotive EngineersSAE AS-4 SAE Aerospace Standard-4
SCC	serial commutation controller
SPAWAR	Space and Naval Warfare Systems Command
SSC-Pacific	Space and Naval Warfare Systems Center – Pacific, Unmanned Systems Branch
STANAG	standardization agreement
TCP	Transmission Control Protocol
TCP/IP	Internet Protocol Suite
TDS	Target Detection System
TRADOC	Training and Doctrine Command
TTPs	tactics, techniques, and procedures
UAS	unmanned air system
UCR	UMS communication repeater
UDP	User Datagram Protocol
UGS	unattended ground sensors
UGV	unmanned ground vehicle
UMS	unmanned system
USV	unmanned surface vehicle
VTOL	vertical takeoff and landing
WGS	World Geodetic System
WiFi	wireless fidelity
wrt	with respect to
XML	extensive markup language